NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

TERRESTRIAL COMMUNICATION BETWEEN WIRELESS SENSOR NETWORKS USING BEAM-FORMING AND SPACE DIVISION MULTIPLE ACCESS

by

Chris Edward Taylor

June 2008

Thesis Advisor:	Murali Tummala
Co-Advisor:	John McEachen

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.			
1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** June 2008	**3. REPORT TYPE AND DATES COVERED** Master's Thesis	
4. TITLE Terrestrial Communication Between Wireless Sensor Networks Using Beam-Forming and Space Division Multiple Access		**5. FUNDING NUMBERS**	
6. AUTHOR(S) Chris Edward Taylor			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000		**8. PERFORMING ORGANIZATION REPORT NUMBER**	
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A		**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited		**12b. DISTRIBUTION CODE**	

13. ABSTRACT (maximum 200 words)

In this thesis, methods for forming a communications link of Wireless Sensor Networks (WSN) by enabling each WSN to act as a smart antenna are presented. Each WSN is simulated as a set of randomly placed sensor nodes within a planar area. The proposed method involves a searching WSN, a receiving WSN and a link budget for establishing the link.

The searching WSN has the task of transmitting a search beam in order to find adjacent WSNs. Like a lighthouse this is done in a rotating beam style search using the sensor nodes as an aperiodic array. Results show that for a random array, we can achieve a specific beamwidth and gain as a function of the number of elements and area. We also demonstrate that for a given required gain level we can spatially thin the array without significant loss of gain or the effects of grating lobes.

The receiving WSN uses a spread spectrum based space division multiple access (SDMA) receiver. This receiver is simulated to determine the direction of arrival from the searching WSN and to extract the location information from the searching WSN's signal with additive white Gaussian noise. From the DOA and the location information within the arriving signal, the WSN has sufficient knowledge to respond to the query of the searching WSN and form the communications link.

14. SUBJECT TERMS Wireless Sensor Network, Space Division Multiple Access, SDMA, Direction of Arrival, DOA, Spread Spectrum, Random Arrays, Smart Antennas			**15. NUMBER OF PAGES** 117
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

TERRESTRIAL COMMUNICATION BETWEEN WIRELESS SENSOR NETWORKS USING BEAM-FORMING AND SPACE DIVISION MULTIPLE ACCESS

Chris E. Taylor
Lieutenant Commander, Untied States Navy
B.S., Rice University, 1985

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June 2008

Author: Chris Edward Taylor

Approved by: Murali Tummala
Thesis Advisor

John McEachen
Co-Advisor

Jeffrey B. Knorr
Chairman, Department of Electrical and Computer Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

In this thesis, methods for forming a communications link between Wireless Sensor Networks (WSNs) by enabling each WSN to act as a smart antenna are presented. Each WSN is simulated as a set of randomly placed sensor nodes within a planar area. The proposed method involves a searching WSN, a receiving WSN and a link budget for establishing the link.

The searching WSN has the task of transmitting a search beam in order to find adjacent WSNs. Like a lighthouse this is done in a rotating beam style search using the sensor nodes as an aperiodic array. Results show that for a random array, we can achieve a specific beamwidth and gain as a function of the number of elements and area. We also demonstrate that for a given required gain level we can spatially thin the array without significant loss of gain or the effects of grating lobes.

The receiving WSN uses a spread spectrum based space division multiple access (SDMA) receiver. This receiver is simulated to determine the direction of arrival from the searching WSN and to extract the location information from the searching WSN's signal in additive white Gaussian noise. From the DOA and the location information within the arriving signal, the WSN has sufficient knowledge to respond to the query of the searching WSN and form the communications link.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

EXECUTIVE SUMMARY

A wireless sensor node is a low cost, small, battery powered electronic device designed to monitor or measure a physical phenomenon of the environment around it, such as temperature, pressure, humidity, soil pH, vibration, motion, light, sound, radiation, and chemical presence. Wireless sensors are generally composed of a microprocessor, the appropriate sensor(s), a transceiver and a power source. Once the sensor makes a detection, the information collected is relayed back to an end user for further processing or analysis. A collection of sensors in close proximity to one another may form a Wireless Sensor Network (WSN) by establishing communications with one another through some form of self-organization. There are many different applications for WSNs, both commercially and militarily. In this work we assume a military application where the sensor nodes are deployed in hostile territory, and therefore delivery of the sensors may be via an aerial vehicle. Delivery in this manner would generally lead to a random spatial distribution of the sensors, yet the sensor nodes are close enough to form a local wireless sensor network. The nodes of this WSN may act collectively as a group for such actions as distributed processing or forming a distributed antenna array.

In this thesis, we assume the deployed WSNs must transmit their information to the end user without access to the delivering aerial vehicle, but rather terrestrially. In the solution proposed here, where the WSNs are deployed to multiple locations nearby to one another, they can form a terrestrial Over The Horizon (OTH) network by forming a multi-hop communication link using the individual WSNs. In order to form this link, the WSNs must also solve the additional task of locating one another. The WSNs act collectively to form a smart antenna array. By establishing communication links between adjacent WSNs, an extended network made up of what has now become nodes of a larger Over The Horizon (OTH) network is formed. Once the OTH network is connected, the individual WSNs become part of a larger network that can relay sensed data to an end user.

The goal of this thesis is to develop and validate methods to establish OTH communication links using wireless sensor networks by enabling each to act as a smart

antenna array. In this thesis, we approach establishing the communication link between a searching WSN and a receiving WSN as a four-step process: localization of the sensor nodes within each WSN to determine its geometry; the search by one WSN for adjacent WSNs by acting as an smart antenna; determination of the direction of arrival and range of the searching WSN by the receiving WSN; and then forming the multi-hop link using WSNs by optimizing the gain of the beam forming the link.

In searching for the adjacent WSN, each of the sensors act in concert as a smart antenna array. To describe how we will form a beam using a group of randomly placed sensor nodes, we review uniform linear and planar arrays then the random planar array. Each WSN is modeled as a uniformly random distribution of sensors nodes. In order to link the WSNs, we will wish to control the beamwidth, gain and direction of the array beam. We show analytically that we can steer the beam in an arbitrary direction. We also demonstrated control over the beamwidth and gain by providing MATLAB results and compare the results to analytical solutions. To demonstrate the search technique, a 40 element, $2.5\lambda \times 2.5\lambda$ random planar array scanning the horizon is presented.

The results show that by choosing the physical size, likely as sub-section of the WSN we can choose the beamwidth of the search array. Within this area, we demonstrate control over the gain by choosing the number of elements employed in the array. Alternately, for a given required gain level we show that we can thin the array (or reduce density) dramatically without significant loss of gain or the effects of grating lobes. It does not matter which particular sensor nodes within the WSN are used as long as, on average, they are randomly distributed within the physical area chosen. This is important as we may wish to distribute the burden of transmitting, and thus consuming energy among the nodes within the WSN.

From the perspective of the receiving WSN, this array needs to know the direction of arrival (DOA) and range to the searching WSN in order to form an optimum link. A new approach to DOA determination using spread spectrum space division multiple access (SDMA) is introduced, demonstrated for a uniform linear array and then validated for a random planar array. We show that the method works better for Walsh-Hadamard

spreading codes than for pseudorandom noise codes and that 64 chips is sufficient to accurately establish the DOA and character of a linear antenna array. Using a random planar array, we determined the DOA of an incident test signal, then simulated extracting a message data stream from this signal in a noisy channel with SNRs of 20 db and -5 dB, accurately extracting the message.

A link budget analysis is performed for a simulation involving two WSNs using random planar arrays on a flat earth. This example uses commercially available sensor motes at two transmit power levels. Gain calculations are made for each power level, solving for the required number of transmit and receive elements in each array. In this example we use the Crossbow MICAz® sensor operating at 2.4 GHz. The path loss provided the greatest amount of uncertainty in the link budget. Of the models examined, the range of values for a 500 m signal at 2.4 GHz varied by approximately 15 dB. After determining the gain of the random planar array, we found that for a 500 meter separation we could use 17 elements on the receive side and 25 elements on the transmit side when all elements were operating at full power. When all were operating at minimum power, we found that using 45 elements to receive and 45 elements to transmit provided sufficient margin.

In this thesis we have developed and validated methods to establish communication links between wireless sensor networks using smart antenna beamforming techniques, spread spectrum space division multiple access as a method for a receiving WSN to determine the direction of arrival and message sent from an adjacent searching WSN and the use of link analysis to optimize the number of sensors used to form the link.

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

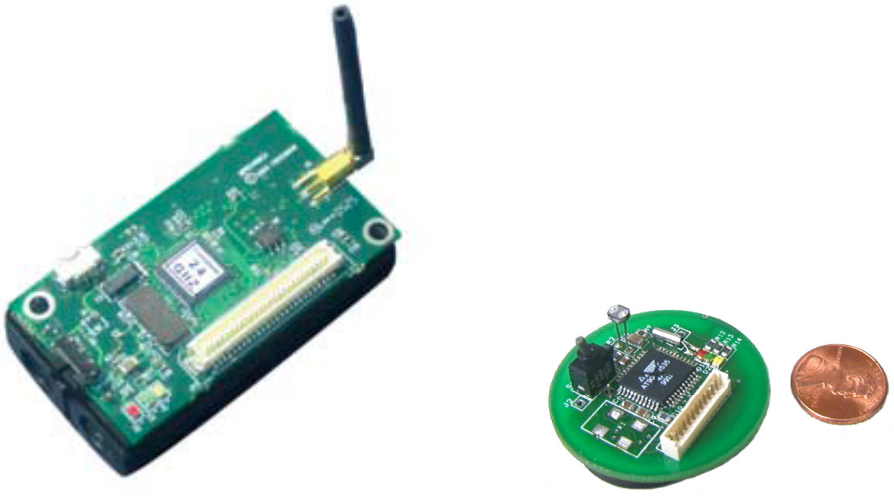
I would like to thank Professor Murali Tummala of the Naval Postgraduate School in Monterey, California for his patience, guidance and contributions to this work.

I would also like to thank my wife Lori for her support and patience during a very busy year. Without her support, this would never have been possible.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A wireless sensor node is a low cost, small, battery powered electronic device designed to monitor or measure a physical phenomenon of the environment around it, such as temperature, pressure, humidity, soil pH, vibration, motion, light, sound, radiation, and chemical presence. Wireless sensors are generally composed of a microprocessor, the appropriate sensor(s), a transceiver and a power source. Once the sensor makes a detection, the information collected is relayed back to an end user for further processing or analysis. Figure 1a shows an example of the MICAz sensor mote by Crossbow® Technologies [1] and in Figure 1b a MICA2DOT sensor by Crossbow®.



(a) MICAz (b) MICA2DOT Sensor Mote

Figure 1. Examples of Sensor Motes from Crossbow®(From [1])

A collection of sensors in close proximity to one another may form a Wireless Sensor Network (WSN) by establishing communications with one another through some form of self organization. This group may act collectively to increase sensing or processing power by performing some of the computations onboard the sensor node itself [2].

Many different types of applications for wireless sensor networks exist, in both industry and the military world. Because they are wireless and self-organize, wireless sensor networks have the potential to relay information without a large infrastructure cost or physical impact. Commercial applications of wireless sensor networks include industrial monitoring, building controls, security, traffic management, weather, wildlife tracking, and agricultural field temperature-sensing networks [3]. Still another application consists of sensors that monitor conditions in the London underground tunnels and water systems [4].

Military applications may include remote sensing of nuclear, biological or chemical (NBC) agents, motion, video, etc. It is assumed that the sensor nodes are deployed in hostile territory and therefore delivery of the sensors may be via an aerial vehicle. Delivery in this manner would generally lead to a random spatial distribution of the sensors, yet the sensor nodes are close enough to form a local wireless sensor network. Once established, the issue becomes how to get the information from the wireless sensor network to the end user. The nodes of this network may act collectively as a group for such actions as distributed processing or forming a distributed antenna array. Vincent [5] and Chan [6] explored the use of beam forming to communicate directly with an overhead UAV as a method to relay information to the end user.

In this thesis, we assume the deployed WSNs must communicate their information to the end user without access to a UAV. In the solution proposed here, where the WSNs are deployed to multiple locations nearby to one another, they form a terrestrial Over The Horizon (OTH) network by linking the individual WSNs. In order to form this link, the WSNs must also solve the additional task of locating one another. Each WSN acts collectively to form a smart antenna. A key assumption to this problem is that the distance between WSNs is greater than the individual range of a single sensor node. Thus the sensor nodes in the WSN must act cooperatively in order to have sufficient transmitting gain to reach the next WSN.

This problem combines and draws from several different disciplines in order to propose a solution. Each of these topics will be discussed further in this thesis. In order to form the initial local network, the concepts of localization of the network are required.

Once the geometry is known, the sensor nodes of the WSN act collectively to form a distributed smart antenna. A smart antenna generally refers to an array antenna that can adapt its beam pattern using a computer processor [7]. Because the sensor nodes are scattered in a random manner, the concepts of random processes are used to apply them in the smart antenna composition. The concepts of spread spectrum and signal processing come into play in the determination of the direction of arrival for the beams connecting two WSNs.

A. DESCRIPTION OF THE PROBLEM

We begin with a description of the wireless sensor network, depicted in Figure 2. Consider a set of wireless sensor nodes, dispersed within an arbitrary boundary. The sensor nodes form an ad hoc network by establishing communications with one another. The boundary is defined as the range limit of the communicating sensor nodes within the ad hoc network.

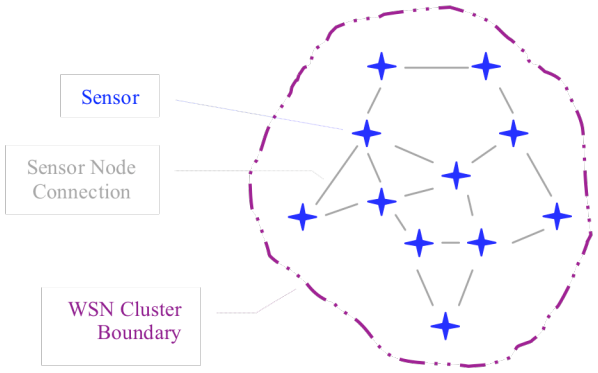


Figure 2. Wireless Sensor Network

The individual sensor nodes may be randomly positioned within a boundary defining the WSN. The goal of this thesis is to explore methods for forming communication links between adjacent WSNs. By establishing communication links between adjacent WSNs, we can form an extended network made up of what has now become nodes of a larger Over The Horizon (OTH) network. The general case is shown

in Figure 3. Once the OTH network is connected, the individual WSNs become part of a larger network that can relay sensed data to an end user.

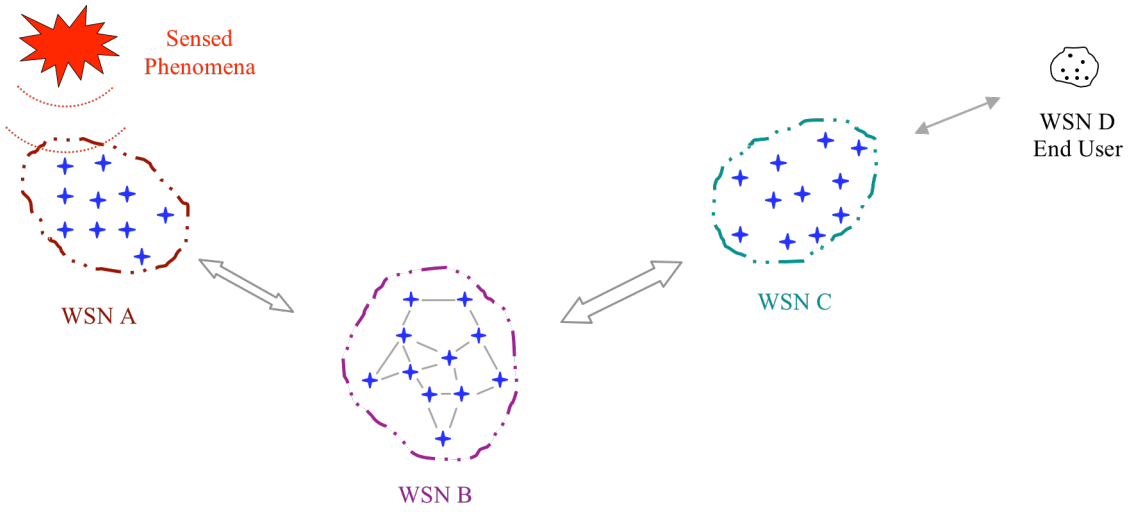


Figure 3. Generalized Over The Horizon Network of Wireless Sensor Networks

Next we consider a tactical military example where we wish to link WSNs to form an OTH network. Consider an Unmanned Aerial Vehicle (UAV) or other aircraft that disperses sensor networks to various locations in hostile territory, as envisioned in Figure 4. The goal is to connect the WSN A to B, B to C, etc. such that we eventually have a link from A to E.

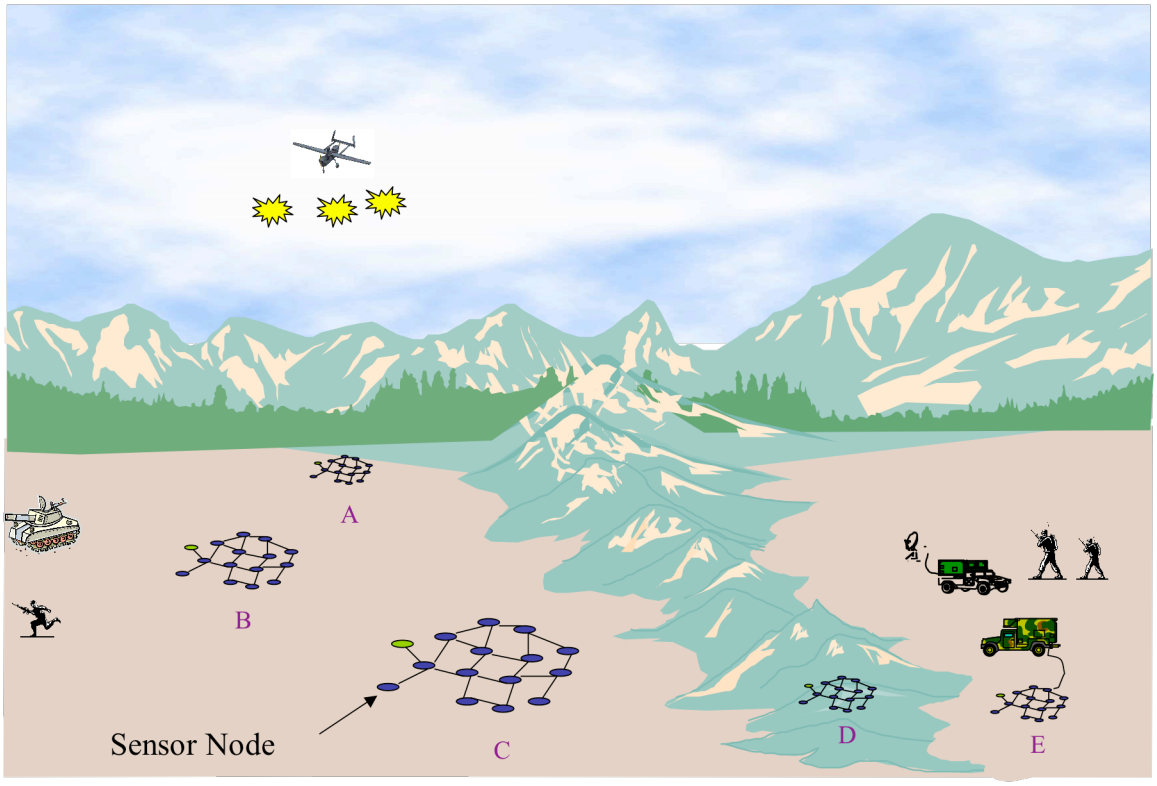


Figure 4. Tactical Application of Linked Wireless Sensor Networks

In previous work related to this scenario, each of the WSNs communicated directly to the UAV when it was available overhead [5]. In this work we seek to communicate not via a UAV, but by establishing links between already placed WSNs.

In more general applications, a WSN need not be on the ground but can be attached to a building, bridge, structure or even a moving vehicle. The goal here is for the WSNs to find each other via beam forming and then establish a communications link at the highest rate possible.

B. THESIS OBJECTIVE

The goal of this thesis is to develop and validate methods to establish OTH communication links using wireless sensor networks by enabling each to act as a smart antenna array.

To begin forming the OTH communication link, we examine methods to form a search beam and transmit this beam around the horizon in order to illuminate an adjacent WSN. We will assume that each WSN is composed of a uniformly distributed random array of sensor nodes. Using theory from periodic and aperiodic antenna arrays, we will develop methods to model and characterize the random array in terms of gain and half power beamwidth as a function of the number of elements and the size of the random array. We will demonstrate using MATLAB a random planar array forming a beam and scanning the horizon.

For the WSN being illuminated, the direction of arrival (DOA) and location of the searching array must be determined in order to form an optimum link. We seek to develop and simulate methods to determine the DOA and location information using a direct sequence spread spectrum based space division multiple access technique applied to the WSNs.

In order for the WSN to act as a smart antenna, the local geometry of each WSN must be determined in order to define the sensor node positions and boundaries. Because this topic has been extensively reported in the literature, only an overview of applicable localization techniques is presented in Appendix A. To establish an optimum communications link, we will examine a pair of WSNs using a link budget analysis. We will determine the gain and number of sensor nodes required for this communication link. Finally, the overall OTH network is realized by establishing a series of communication links as described above. This area is present conceptually only, as the focus of this thesis is to investigate the establishment of the link between two adjacent WSNs.

C. RELATED WORK

The concept of beam forming using an array of randomly distributed sensor nodes has been explored previously in several works. Tong [8] studied random arrays in the context of randomly distributed micro-strip patch antennas placed on the sides of ships acting as a digital phased array radar aperture. Batson [9] studied wireless sensor networks as a means to collect signals intelligence by building adaptive arrays. Vincent

[5] considered a similar problem to the one we explore in this thesis, namely that of a WSN deployed in a remote location that uses adaptive beamforming to communicate to an overhead UAV. Vincent also explored energy burden sharing across the sensor network, a concept that is conceptually useful to an over the horizon network. Chan [6] studied adaptive beamforming for direction of arrival calculation as well as the effects of position error and element failure on the beam of a random array of elements. Godara [10] presents an extensive compilation of techniques for beamforming and direction of arrival estimation, as well as a comparison of the relative performance of each. While Hong, et al [11] use cooperative methods to establish a relay network of sensors, they use spatial diversity techniques rather than beamforming.

This thesis builds upon the concept of the random planar array of sensor nodes, but here the objective is to build an over the horizon network of WSNs. We also propose new ways for searching, tracking and establishing a link between adjacent WSNs. The communication links between adjacent WSNs are established by using beam searching and direction of arrival methods based upon spread spectrum techniques.

D. THESIS ORGANIZATION

This thesis is organized as follows. Chapter II provides a link budget analysis to determine the gain requirements for the transmitter and receiver given data rates, range and power considerations. Chapter III describes the use of beam-forming as a method to search for adjacent WSNs. Simulation results for random planar arrays and search patterns are included. Chapter IV describes the link establishment process and the determination of the direction of arrival (DOA) using spread spectrum based space division multiple access (SDMA) methods. Simulation results of a random planar array detecting the arrival of a signal of interest as well as extracting a sample data stream are presented. Chapter V presents the conclusions of this work and topics for further research. Finally, Appendix A summarizes the sensor node localization methods and Appendix B includes MATLAB code used in this thesis.

THIS PAGE INTENTIONALLY LEFT BLANK

II. COMMUNICATIONS LINK: LINK BUDGET CONSIDERATION

The over the horizon (OTH) communications system is essentially a network of networks linking the individual WSNs. Several issues must be resolved in order to link these WSNs of an OTH network. This thesis examines the link between just two of the WSNs, but for completeness, we discuss the bigger issues of the OTH network as a whole. Many of these issues are not too different from the issues discussed in Appendix A on localization. In this chapter, we begin with a conceptual look at the OTH network, then consider just two WSNs of the overall chain of WSNs that eventually form the OTH network. Within this view, we consider them as two points of a communications link and apply a link budget analysis.

A. BUILDING THE OVER THE HORIZON NETWORK

Each WSN of the OTH network must find the adjacent WSN to connect to. Once the initial link is established, say from A to B as in Figure 3, WSN B must now continue the link and connect to a third WSN, for example WSN C of Figure 3. While establishing the second link the WSN node must still maintain contact with the first WSN. A particular WSN may subdivide as a method to establish contact with more than one other WSN. The problem of subdividing an individual WSN may be addressed in several ways. Perhaps the WSN could physically partition the sensor nodes to address and connect to other nodes, e.g. the Eastern most nodes form an array to communicate to another WSN to the East, and the Western most sensor nodes form an array to communicate to the West, as depicted in Figure 5. In Figure 5, we show WSN B subdivided laterally, such that the red nodes form an array and communicate with WSN A and the green nodes form an array and communicate with WSN C.

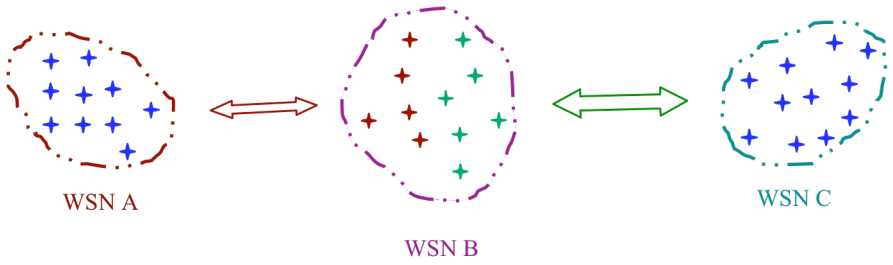


Figure 5. WSN B Sensor Nodes Partitioned to Simultaneously Communicate With WSN A and WSN B

Alternately, the WSN could employ some subset of the sensors using a multiplexing technique based on time, code or space. Additionally, the frequency band could be split into different frequency channels for communicating to individual surrounding WSNs. This thesis does not address the details of subdividing the WSNs, but rather approaches this part of the overall process conceptually.

In the following chapters, we will model the transmitting and receiving WSNs as smart antennas, taking advantage of the element localization information, the random nature of the element spatial distribution to form a transmit beam and a spread spectrum SDMA technique to determine the relative geometries between WSNs.

Essentially, the WSN will now be functioning as individual antennas, and thus the communication link can broadly be described by Figure 6. This figure depicts a terrestrial communications link formed over some distance R between antennas A and B. Because of the nature of the WSNs, we can change some of the characteristics of the communication link, such as its effective aperture and the number of elements used to form the antenna. This chapter seeks to examine those parameters given the known parameters such as range, required data rate, path loss and available transmitter power. We begin with a discussion of the link budget equation and then perform an analysis using a commercial sensor mote and an example random planar array.

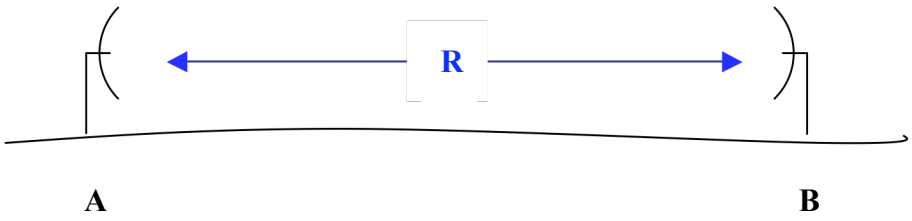


Figure 6. Simplified Terrestrial Radio Communications Link

B. LINK BUDGET EQUATION

The link budget allows us to perform a tradeoff analysis when we are given control over components of the link. In this section, we examine the individual terms of the link budget equation [12]:

$$M_{dB} = P_t + G_t + G_r - \left(\frac{E_b}{N_o}\right)_{\text{Re}\,qd} - R_b - kT_s - L_S - L_O \qquad (1)$$

where M_{dB} is the link margin (dB), P_t is the transmitter power (dBW), G_t is the gain of the transmitting array (dBi), G_r is the receiving array gain (dBi), E_b / N_0 is the energy per bit to noise power spectral density ratio (dBW/Hz), R_b is the bit rate of the communications between WSNs (dB-bit/sec), k is Boltzman's constant (1.38×10^{-23} J/K), T_s is the effective system temperature (K°) generally given as $T_s = T_a + T_e$,where T_a is the antenna noise temperature and T_e is the receiver noise temperature. The path loss, L_s (dB) is the loss of a signal over the distance from point A to point B, including multipath, fading, two ray bounce, weather, etc. and L_o includes "other losses" such as transmitter inefficiencies, line loss, polarization mismatch, etc. [12]. The link margin is the difference between what we need to establish the link and what we have available from the combined components of the communications link. We assume that each transmitter can vary its power output in discrete steps, but that all elements transmit at equal levels. The gain of the transmitting and receiving arrays refer to the gain of the random planar array of judiciously selected elements. The gain of a random array of transmitters will be described in Chapter III on random arrays and will be one of the key parameters we can vary. Here, we will assume the "other losses" are negligible.

1. Link Analysis

Here we consider two WSNs deployed into separate but nearby regions on a flat environment, not unlike the scenario depicted in Figure 4. For example, let us consider as sensors the Crossbow® MICAz, a commercially available sensor mote, to apply to this scenario. MICAz operates at 2.4 GHz and generates RF transmitter power from 40μW to 1mW. The nominal outdoor range for a single receiver/transmitter pair is given as 100m at the receive sensitivity of -120 dBW [13]. The maximum data rate for this mote is 250 kbps. Using devices like these one may create a WSN containing 50 sensors arrayed over a 5 m $\times$ 5 m ($40\lambda \times 40\lambda$) region. For simplicity, we will assume that both WSNs are the same size and contain the same number of elements and that the separation distance between the two WSNs is 500 m (4000 λ).

We will assume that all 50 of the receive sensor nodes are awake and in "receive" mode, although generally this need not be the case. We now evaluate the terms in the link budget equation for this example. The digital modulation scheme for the MICAz is BPSK or OQPSK per the IEEE 803.15.4 standard. As a first example, we assume that each transmitter is operating at the maximum power level of 1 mW which equates to -30 dBW. At minimum power, the transmitters produce 40 μW or -54 dBW which we consider second. We assume that all the transmitters operate at the same power level. For the path loss term of the link budget, the contribution is not as clear. Before continuing with the link equation, we briefly examine the path loss for two sensor nodes positioned on the ground.

a. Path Loss Models

Many models exist for land mobile radios and wireless LAN (WLAN) systems and produce a wide range of values when applied to sensor nodes. A model to estimate path loss for sensor nodes at ground level is not readily available as most address antenna heights of approximately 1 m or above. Models that do include antenna height are relatively sensitive to the height of the transmit and receive antennas. For the MICAz sensor mote, which uses a one-half wavelength dipole (~6.25 cm) and sits on a

battery pack and circuit board (~3 cm), this gives us an approximate antenna height of 10 cm. We now examine the most common models for path loss.

The free space path loss at a distance d and wavelength λ is given by [12]:

$$L_s = 10\log_{10}\left(\frac{4\pi d}{\lambda}\right)^2$$

The free space path loss is not applicable to the scenario under consideration, but it serves as a baseline for comparison with the other models that will be presented below. In this case, the antennas are very close to the ground and therefore the path loss is expected to be greater. Green and Obaidat developed a path loss model for wireless LANs operating at 2.4 GHz that takes antenna height into account and was experimentally studied under conditions similar to ours. The Green and Obaidat path loss equation is given by [14]:

$$L_{WLAN} = 7.6 + 40\log_{10} d - 20\log_{10} h_t h_r \quad (dB)$$

where d is the separation distance and h_t and h_r are the heights of the transmit and receive antennas, respectively.

Okumura [15] and Hata [16] produced landmark papers on propagation loss in land-mobile radios, but these models are for signals below 1500 MHz and generally for antenna heights of 1 m or more. The Green and Obaidat models demonstrated very good agreement with their experimental results, for antenna heights of 1-2 m, but produce higher losses when used for antenna heights of 5-10 cm.

A propagation model based on a flat Earth assumption, where the Earth is represented as a dielectric is another approach. Linnartz [17] provides the path loss at a distance d for the flat Earth as:

$$L_{pE} = \frac{\lambda^2}{(4\pi d)^2}\left[2\sin\left(\frac{2\pi}{\lambda d} h_t h_r\right)\right]$$

where λ is the wavelength, and h_t and h_r are the heights of the transmit and receive antennas, respectively. Because the antennas are so close to the ground and we are using a 2.4 GHz example this model would only apply in a very limited number of cases where the environment was free of scattering objects.

The calculated path losses for the 2.4 GHz signal at a separation of 500 m with both antennas at 10 cm are listed as follows:

Free Space	94 dB
Hata Model	121 dB
Green/Obaidat Model	134 dB

Based upon the operating environment, the path loss for this system ranges from 94 to 134 dB. Each of the models, except Free Space, is sensitive to antenna height. The Green and Obaidat model seems reasonable for use in this work given that their model was developed specifically for WLAN applications at 2.4 GHz and had good agreement with experimental results. However, this model used much greater antenna heights suggesting that using this model for sensor nodes on the ground may not produce accurate results.

b. Link Budget Gain Calculations

We wish to know what level of gain is required for the transmit and receive antenna arrays forming the communications link shown in Figure 6. We can solve the link equation to determine, with margin *M,* what gain the two WSN arrays must produce. Since we are given the receive power sensitivity P_r, of the MICAz, assumed to incorporate the data rate and required E_b/N_o, Equation (1) simplifies to:

$$M_{dB} = P_t + G_t + G_r - P_r - L_s \tag{2}$$

where M_{dB} represents the difference between the actual received power and the minimum useable power, given by the receive sensitivity. Also note that the total power transmitted by the array of N_t transmitting elements is $N_t P_t$.

Using Equation (2) and combining the terms from the link analysis and the Green/Obaidat path loss estimate, we have for the 1 mW (-30dBW) transmitter:

$$M = (G_t + G_r) + N_t + P_t - P_r - L_s$$
$$M = (G_t + G_r) + N_t - 30 - (-120) - 134)$$
$$(G_t + G_r) = M - N_t + 44$$

Thus the combined gains of the transmitting array and receiving arrays must meet a margin balanced by the total transmitted power less a factor for path loss.

Excluding the free space model, the range of path loss values is on the order of 15 dB, suggesting that we should choose this as our margin.

When the transmitters are operating at minimum power (-54 dBW), from Equation (2) the required gain is:

$$(G_t + G_r) = M - N_t + 68$$

In this case, it is clear that we must now either decrease our acceptable margin or increase the number of transmitters or both in order to form the communications link. Because transmission consumes much more energy than reception, it is envisioned that we will use many more receive nodes than transmit nodes to meet the gain requirement. For example we may use 15 transmit and 30 receive nodes. In Chapter III, we will be able to determine these numbers.

Clearly the path loss dominates the overall margin requirement. One method to reduce the path loss is to elevate the sensor nodes. This increases the antenna height and rapidly reduces the path loss, thus leading to greater range. Typically, doubling the antenna height increases the received power by a factor of 4 [17]. For example, the sensor network could be placed on the side of a building or rooftop, elevated on top of a vehicle, attached to lamp posts, etc.

In summary, we applied the link budget equation to a link formed between two WSNs and determined the required combined gain from the receive and transmit antennas. In the next chapter we examine how to calculate this required gain as well as some other parameters of the antenna we will form from the WSN.

THIS PAGE INTENTIONALLY LEFT BLANK

III. THE SEARCH FOR OTHER WIRELESS SENSOR NETWORKS

In this chapter, we approach the communications link from the viewpoint of a searching WSN in the eventual OTH network. For example, this could be WSN A of Figure 3, which has sensed a phenomenon and seeks to pass this information along to the end user. This WSN has the task of transmitting a search beam in order to find adjacent WSNs. Like the rotating beam of a lighthouse, the WSN will steer its beam in a rotating manner around the horizon by using the local network of sensor nodes as an aperiodic antenna array.

In order to address the method, we first discuss the concepts of periodic arrays. Then we extend these concepts to the random array, a type of aperiodic array. Next we explore the characteristics of the two dimensional random array, using MATLAB to calculate gain, beamwidth and sidelobe levels. Finally, we discuss the issues of the search between WSNs, using random planar arrays.

A. BEAM FORMING WITH PLANAR ARRAYS OF RANDOMLY PLACED SENSOR NODES

To describe how we will form a beam using a group of randomly placed sensor nodes, we begin by reviewing the theory of the uniform linear array (ULA) and the uniform planar array. From this, we make the extension to planar arrays containing randomly placed elements. In each case, we assume that the sensor antenna elements are half wave dipoles oriented vertically to the plane.

1. Geometric Arrays

The far field pattern for an array of geometrically arranged antennas is calculated using pattern multiplication; that is, the far field radiation pattern $F(\theta)$ can be expressed as the product of the element pattern $E_0(\theta)$ and the array factor $AF(\theta)$ [18]:

$$F(\theta) = E_0(\theta) \times AF(\theta)$$

For a $\hat{z}$-directed, half wavelength dipole, the normalized element factor is [18]:

$$E_o(\theta) = \frac{\cos\left(\frac{\pi}{2}\cos\theta\right)}{\sin\theta} \tag{3}$$

and is assumed to be the element in all the gain pattern calculations that follow. The array factor characterizes the radiation pattern based upon the geometry of the antenna elements. We now discuss the one-dimensional uniform linear array.

a. Uniform Linear Array (ULA)

Figure 7 shows the geometry of a uniform linear array and its coordinate system, and observation direction *R(θ)*. The angle θ is measured from the z axis.

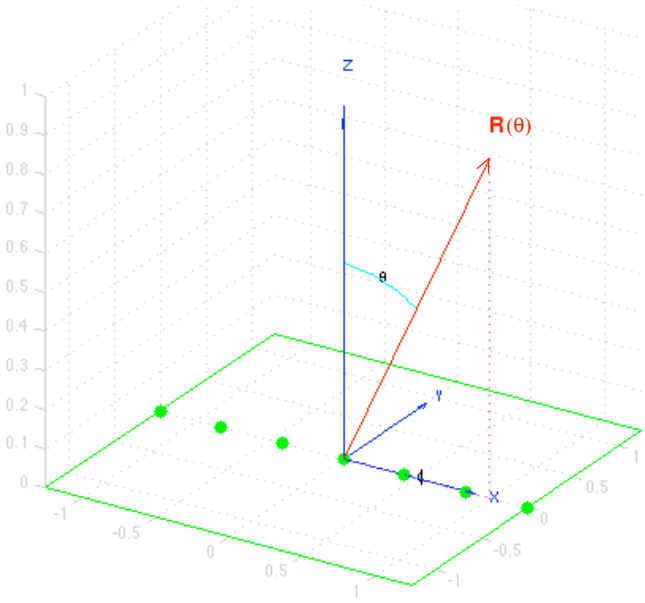


Figure 7. Uniform Linear Array Geometry and Coordinate System

For an equally spaced set of N isotropic radiators with wave number $k = 2\pi / \lambda$, spaced by a distance d along the x axis, the array factor may be expressed as: [18]

$$AF(\theta) = \sum_{n=0}^{N-1} A_n e^{jnkd\sin\theta} \tag{4}$$

where A_n is the amplitude of the drive of the n^{th} current element, and $e^{jnkd\sin\theta}$ is the phase contribution of n^{th} element. Note that the array factor is rotationally symmetric around the x-axis. Figure 8 illustrates how the observation angle θ and the spacing d affect the phase term of the array factor. For example, when θ is zero, all the elements have the same phase because $\sin(\theta)$ is zero; when $\sin(\theta) = 1$, the phase contribution from each element is equal to kd. From this, we see that the spacing, observation angle and wave number determine the phase of the far field array factor.

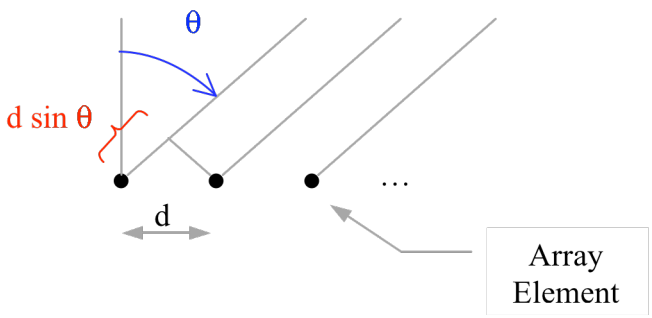


Figure 8. Phase Difference is determined by θ and spacing d

We can simplify Equation (4) by letting

$$\psi = kd\sin\theta$$

and letting all the current amplitudes be equal, i.e.,

$$A_0 = A_1 = \ldots = A_{N-1}$$

Thus, Equation (4) becomes:

$$AF(\psi) = A_0 \sum_{n=0}^{N-1} e^{jn\psi} \tag{5}$$

For N elements, and the beam pointing in the $\theta = 0$ direction, the array factor reduces to

$$AF(\psi) = A_0 N$$

Thus, the maximum array factor is proportional to N.

It can be shown [18] that Equation (5) can be expressed as the more common expression for the normalized array factor f, or pattern factor of a uniformly excited, equally spaced linear array, centered at the origin:

$$f(\psi) = \frac{\sin(N\psi/2)}{N\sin(\psi/2)} \tag{6}$$

A close examination of Equation (6) reveals some important characteristics of array antennas [18]:

- As the length of the array (*Nkd*) increases, the width of the main lobe decreases, i.e., the half power beamwidth (HPBW) becomes smaller.
- As *N* increases, the side lobe levels decrease. The Side Lobe Level (SLL) is expressed as the ratio of the maximum value of the largest side lobe to the maximum value of the main lobe. This is usually described in the literature as "dB below the main lobe."

A characteristic of periodic arrays is revealed when the spacing between elements increases beyond a certain level, and energy from the array is directed into unintended directions. These unintended beams from the array essentially reduce the gain of the array along the desired direction by putting energy into unwanted directions. These unintended beams are called *grating lobes* and are the subject of the next section.

b. Grating Lobes

Grating lobes occur when element spacing is greater than one-half wavelength [18]. These lobes are often of equal intensity to the main lobe(s) of the array and are due to the periodicity of the element spacing. Grating lobes do not occur for arrays with spacing less than one-half wavelength; thus, nearly all periodic arrays conform to this spacing constraint [18]. In order to demonstrate the occurrence of grating lobes, Figure 9 shows the pattern factor for an eight element ULA, with a spacing of 0.5, 1.0 and 1.25 wavelengths between elements. Only the first quadrant is shown, as the pattern is symmetric about 0°. The blue line shows the pattern factor for an array with one-half wavelength spacing and no grating lobes. The red and green lines show the

pattern factor for arrays with a spacing of 1.0 and 1.25 wavelengths and the resultant grating lobes. Notice that for the one-half wavelength spaced array, shown in blue, most of the energy of the array is concentrated in the main beam, while for the larger spacing the energy is dispersed into the grating lobes.

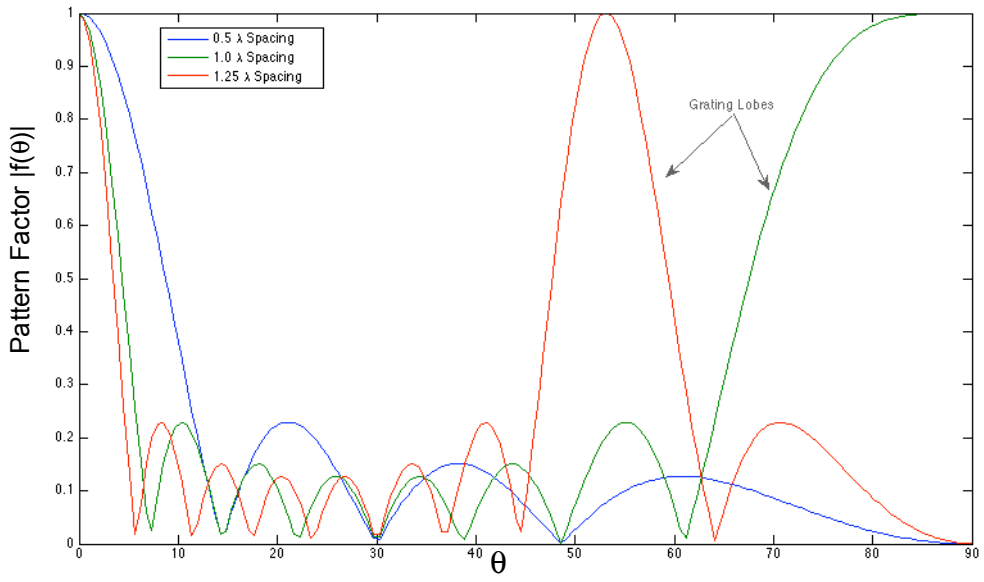


Figure 9. Pattern Factor for an 8 Element ULA Showing Grating Lobes as a Result of Spacing Greater Than One-Half Wavelength

Thus the designer is presented with a dilemma – in order to have a small beamwidth, the ULA must be physically large, but increasing the length will necessarily result in either a higher complexity due to the greater number of elements or, if spread out too much to keep the number of elements down, grating lobes. It will be shown later that since random arrays do not have a periodicity, grating lobes generally do not occur.

We now extend the concepts of the uniform linear array to form a two-dimensional periodic array.

c. Two-Dimensional Periodic Arrays

Figure 10 shows the geometry for a two-dimensional periodic array. Elements are geometrically arranged in the *x-y* plane, and the far field observation

direction is shown by $R(\theta,\phi)$. For this array, the principle of pattern multiplication is also applied. Here we can write the far field pattern $F(\theta,\phi)=F_x(\theta,\phi)\times F_y(\theta,\phi)$, where $F_x(\theta,\phi)$ is the pattern of a linear array along the *x* axis, and $F_y(\theta,\phi)$ is the pattern of a linear array along the *y* axis [19].

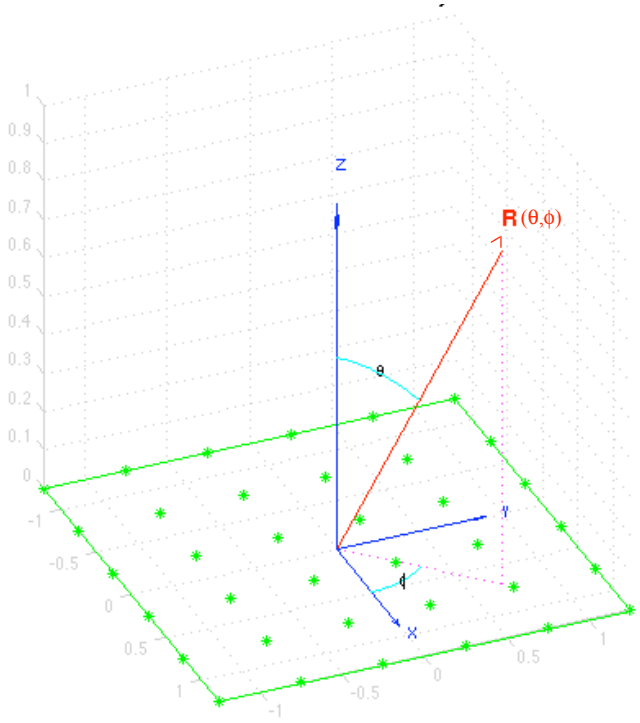


Figure 10. Uniform Two Dimensional Array Geometry and Coordinate System

The array factor for a two dimensional periodic array with *M* elements of spacing d_x in the *x* direction and *N* elements of spacing d_y in the *y* direction is given by [20]:

$$AF(\theta,\phi)=\sum_{m=1}^{M}\sum_{n=1}^{N}A_{mn}e^{jk(m-1)(d_x\sin\theta\cos\phi+\beta_x)}e^{jk(n-1)(d_y\sin\theta\sin\phi+\beta_y)} \tag{7}$$

Combining the exponential terms yields:

$$AF(\theta,\phi)=\sum_{m=1}^{M}\sum_{n=1}^{N}A_{mn}e^{jk[(m-1)(d_x\sin\theta\cos\phi+\beta_x)+(n-1)(d_y\sin\theta\sin\phi+\beta_y)]} \tag{8}$$

where A_{mn} is the current distribution at the $(m,n)^{\text{th}}$ element. The array can be steered to an arbitrary direction (θ_o,ϕ_o) via setting [20]:

$$\beta_x = -kd\sin\theta_o\cos\phi_o \text{ and } \beta_y = -kd\sin\theta_o\sin\phi_o \quad (9)$$

When combined with a specific radiation element, the total radiation pattern is found by multiplying the element pattern $E_0(\theta,\phi)$ with the two dimensional array factor $AF(\theta,\phi)$ [18]:

$$F(\theta,\phi) = E_0(\theta,\phi)\times AF(\theta,\phi) \quad (10)$$

The element pattern is the radiation pattern of a single element in θ and ϕ. In our case, the radiation pattern for a half wavelength dipole given is rotationally symmetric in the angle ϕ; therefore, Equation (3) can be used in Equation (10) as $E_0(\theta,\phi)$.

It can be seen that with half-wavelength spacing, even a modest 7×7 wavelength array would contain nearly 50 elements. It is possible to thin out these arrays yet still have control of the beamwidth, gain and grating lobes. One can thin the arrays by placing the elements randomly within a given area at a much lower density than that of a periodic array. The next section develops these concepts.

2. Random Arrays

A random array is an array antenna whose elements are no longer defined by a fixed geometric spacing but rather the element locations are now random variables. Figure 11 depicts the periodic linear array, the random linear array and the random planar array. A one-dimensional periodic linear array has a fixed inter-element spacing, while the element spacing in a random linear array is a random variable. The elements are randomly placed along both the *x* and *y*-axes in a random planar array, i.e., the inter-element spacing is random along both axes.

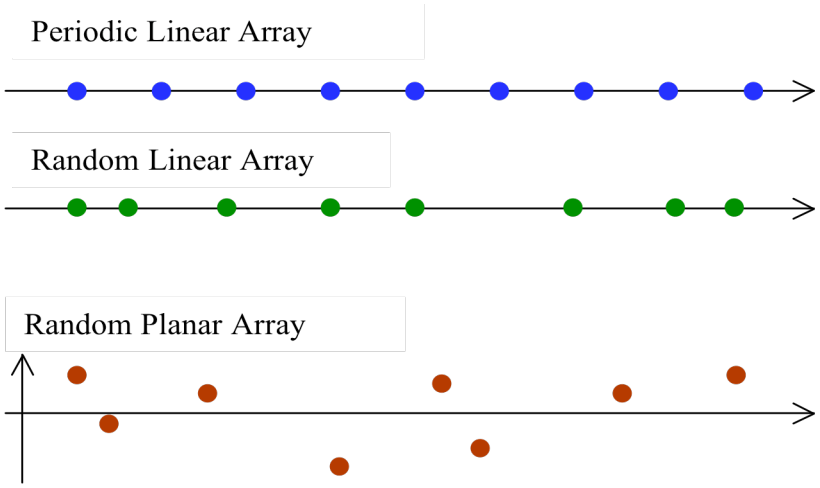


Figure 11. Illustration of Periodic Linear Array, Random Linear Array and Random Planar Array Showing Example Element Distribution

The array factor for a random linear array is similar to the array factor for a periodic array, repeated here for convenience:

$$AF_0(\theta) = \sum_{n=0}^{N-1} A_n e^{jknd\sin\theta}$$

except that now the element spacing term in the exponential is a random variable. For the purposes of performance comparison, the periodic array is termed the *design array* and its array factor is the *design array factor* $AF_0(\theta)$. The phase contribution from each element in a random linear array is no longer a deterministic term, i.e., nd_x, but rather from the actual x coordinate of the n^{th} element, x_n, and thus the phase is also a random variable. The array factor for linear random array becomes:

$$AF(\theta) = \sum_{n=0}^{N-1} A_n e^{jkx_n\sin\theta} \tag{11}$$

Because the element location is now a random variable the array factor is also a random variable. It is customary to represent the array factor of a random array in terms of the ensemble average, referred to as the *average array factor*, given by:

$$\overline{AF(\theta)} = E\left[AF(\theta)\right]$$

where $E[\bullet]$ is the expectation operator.

As two dimensional, or planar arrays, are central to the focus or this work, we extend further the random linear array to the random planar array. In the same manner as with the linear arrays, the inter-element spacing is no longer deterministic, but random in both coordinates. In the previous discussion about periodic planar arrays, the array size contained $N \times M$ total elements, but now we express the total number of elements as simply N. The double sum in Equation (8) is merely a convenient way to express the sum over the two dimensions. Here, it is a single sum because we explicitly step through the coordinates of each element in a random planar array. Thus, for N total elements, the array factor becomes:

$$AF(\theta,\phi) = \sum_{n=0}^{N-1} A_n e^{jk[(x_n \sin\theta\cos\phi+\beta_x)+(y_n \sin\theta\sin\phi+\beta_y)]} \tag{12}$$

Substituting

$$\beta_x = -kx_n \sin\theta_o \cos\phi_o \text{ and } \beta_y = -ky_n \sin\theta_o \sin\phi_o \tag{13}$$

into Equation (12) gives the array factor for the N element random planar array:

$$AF(\theta,\phi) = \sum_{n=0}^{N-1} A_n e^{jk[(x_n \sin\theta\cos\phi-x_n \sin\theta_o \cos\phi_o)+(y_n \sin\theta\sin\phi-y_n \sin\theta_o \sin\phi_o)]} \tag{14}$$

where (θ_o,ϕ_o) determine the beam pointing angles. This equation is used to determine the array factor in all subsequent calculations involving random planar arrays.

Random arrays exhibit some special characteristics that differentiate them from periodic arrays. Some details of random arrays are presented in the following subsections. We will focus our attention on the array performance characteristics of grating lobes, the average array factor, main lobe and beamwidth, average side lobe level and peak side lobe level.

a. Grating Lobes

Grating lobes are a result of the periodic spacing of the elements of the array; there is a coherent buildup of elemental signals at angles other than the steering angle. It should be no surprise then that grating lobes, an artifact of the periodicity of uniform arrays, are significantly reduced in random arrays because of the random nature of the elements [19].

b. Main Lobe and Beamwidth

The shape of the main lobe (maximum gain and HPBW) and the shape of nearby side lobes are not significantly affected by the random nature of the elements [19]. This can be shown by examination of the expression for the average power pattern of a linear random array:

$$\overline{|f(\theta)|^2} = |f_o(\theta)|^2 \left(1 - \frac{1}{N}\right) + \frac{1}{N} \tag{15}$$

where $|f_0(\theta)|^2$ is the design array power pattern and N is the number of elements. More precisely, near the main beam where the normalized power pattern of the design array is approximately one ($|f_o(\theta)|^2 \sim 1$), the main lobe character of the random array (maximum gain, half power beam width) is approximately that of the periodic array. Furthermore, the variance of the array factor is given by [19]:

$$\sigma_N^2(\theta) = \frac{1}{N}\left(1 - |f_0(\theta)|^2\right)$$

where $f_0(\theta)$ is the design array factor and N is the number of elements. As the number of elements N grows, in the limit the variance approaches zero:

$$\lim_{N\to\infty} \sigma_N^2 = 0$$

This simply states that the larger the number of elements, the smaller the variance, and thus the closer the power pattern of the random array is to the design array power pattern.

The broadside half power beam width (HPBW) of a linear geometric array is given as [18]:

$$HPBW = 0.886\frac{\lambda}{L} rad = 50.8^\circ \frac{\lambda}{L} \tag{16}$$

We wish to compare how the HPBW of a random array compares to the HPBW of a design array of the same length.

Figure 12 graphically shows how Equations (15) and (16) compare to the power pattern and HPBW of a linear array – the design array; shown here for arrays of 5, 8 and 20 elements with fixed inter element spacing. As expected, the power pattern of the random array is almost the same as the design array in the vicinity of the main beam,

but forms a "pedestal" of magnitude *1/N* as evidenced in Equation (15). The random nature of the array will cause the average side lobe level to be equal to the pedestal. We also see that in the vicinity of the main lobe, the HPBW is approximately equal to the design array. The approximation improves as the number of elements increases.

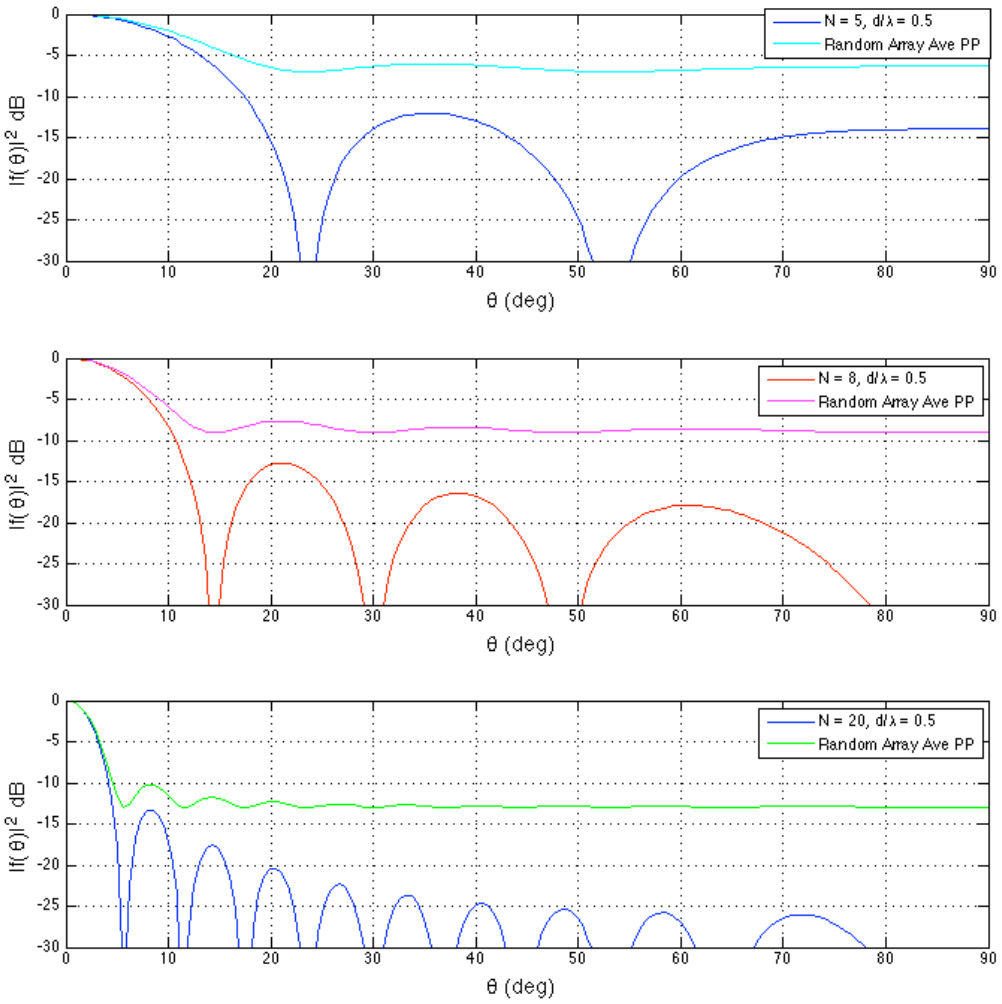


Figure 12. Comparison of the Broadside Power Pattern of 5, 8 and 20 Element Linear Array and Average Broadside Power Pattern of the Same Sized Random Linear Array

The results of Figure 12 are important because they give us an envelope for the expected power level of the random arrays to be developed to build the proposed communication link between WSNs. We also see that as the number of elements

increases, the HPBW of the random array approaches that of the linear array. This is important as we can then easily estimate the HPBW of a random array based on using a simple formula for an N element linear array. Therefore, the half power beam width of the main lobe, for either a random array or a uniform linear array with equal spacing and current is given by Equation (16).

c. Average Side Lobe Level

While the main lobe occurs in the direction the array has been steered, the side lobes are those that occur in all other directions. The arithmetic mean of the peaks of these side lobes is the *average side lobe level.* In contrast to the structure of the main lobe, the distant side lobe structure is affected by randomness of the element spacing. This can be seen from the array factor in Equation (14), where x_n and y_n are random variables, thus the phase is a random variable. Off of the main beam, where the power pattern of the design array is much less than one ($\left|f_o(\theta)\right|^2 << 1$), we see that from Equation (15) the power pattern of the random array is dominated by the *1/N* term. Since the maximum of the power pattern of a random array is equal to that of the linear array, and the average side lobe level of a random array is 1/*N*, we can say that the power in the average side lobe level is equal to *1/N* below the main lobe [19]. In Figure 12 we can see this graphically. All plots are normalized to 0 dB. In Figure 12 we see the average sidelobe level for the 20 element array is approximately $1/N = 10\log_{10}(1/20) = -13$ dB down from the main lobe.

d. Peak Side Lobe Level

While the maximum of all side lobes of a periodic array is generally termed the *side lobe level* [18], for the case of the random array all of the sidelobes are random in nature and average to 1/*N*. The peak side lobe level refers to the maximum peak sidelobe value relative to the average side lobe level [19]. The topic is not easily summarized; however, a rule of thumb for the peak side lobe level is that it is unlikely to exceed the average side lobe level by 10 dB [19].

In conventional periodic arrays, an increase in performance can be achieved by using a non-uniform current distribution across the elements of an array. For a given number of elements, by tapering the current amplitude near the edges of an array, the side lobe level of an array can be reduced and the main beam width increased [18]. For random arrays, however, this is not the case; a tapered current distribution has only a slight impact on the side lobe level [19]. Therefore, in our simulations we will use uniform current amplitude at all nodes of the antenna array.

Having discussed the characteristics of periodic and random arrays, we now proceed to demonstrate this theory using a MATLAB simulation of the random planar array.

B. MATLAB SIMULATION RESULTS – RANDOM PLANAR ARRAY

In this section, the periodic planar array with fixed element spacing is extended to uniformly distributed, randomly placed elements over a planar area. The elements are *z* directed half wave dipoles. The array size is characterized in terms of wavelengths and the number of elements *N*, which represents the *total* number of elements. This is in contrast to periodic planar arrays which commonly list the number of elements along each axis, e.g., $N_x = 5$ and $N_y = 5$ for a total of 25 elements. Here, we simply use $N = 25$ to indicate that there are 25 elements within the array.

Elements in the random planar arrays are restricted to being no closer than $1/10^{th}$ of a wavelength. This is done for two reasons, to better model a physical sensor node as it is unlikely that one would sit on top of another and also to reduce the influence of mutual coupling between antenna elements. Although at a separation distance of $1/10^{th}$ of a wavelength, there is still a contribution from mutual coupling [18]; in this work it is ignored.

In Figure 14, we show the 3 dimensional gain pattern of a random planar array of 25 vertically oriented dipoles with the elements constrained within a $3\lambda \times 3\lambda$ area. This plot is presented to demonstrate the pattern of the random planar array in the upper

hemisphere, with the array beam steered to $\theta = 90°$ and $\phi = 0°$, here oriented to the lower left. Higher gain is shown in bright red, while lower gain is shown in purple and blue. In this thesis, we are interested in forming the terrestrially bound communication link and thus we will focus on the portion of the pattern that lies in the *x*-*y* plane.

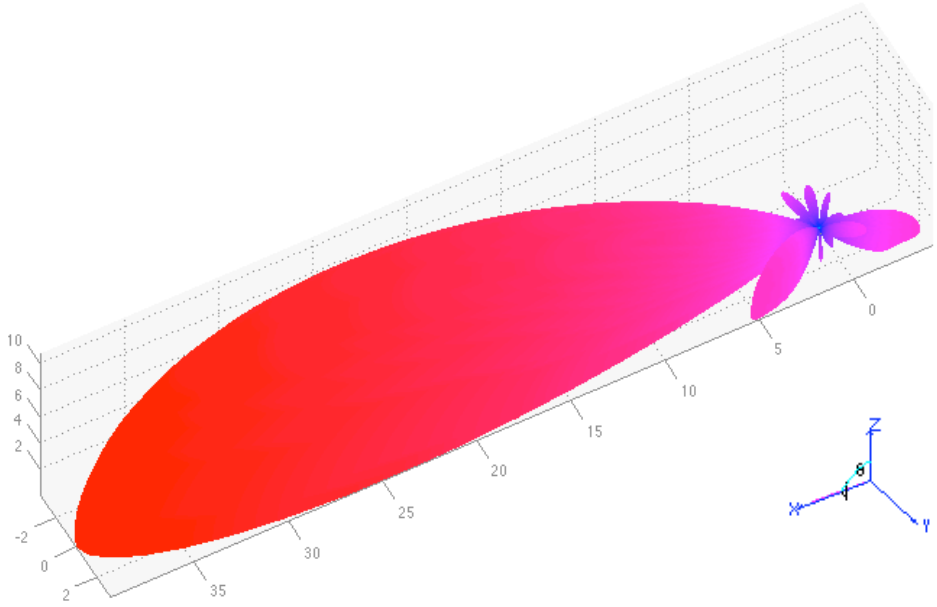


Figure 13. Three Dimensional Gain Pattern of 25 Element $3\lambda \times 3\lambda$ Random Planar Array Steered to $\theta = 90°$ and $\phi = 0°$.

In the following sections, through simulation we examine the relationships between the physical array size and the resulting half power beamwidth; the gain of a fixed size array and the number of elements contained within it; and the gain of the array with a fixed number of elements versus the array size. These results will be useful for developing the proposed communications link between WSNs in later chapters. Note that since the WSNs are all assumed to be on the ground, the search will be along the horizon, or in our simulation's coordinate system (see Figure 10) we have $\theta = 90°$, and ϕ scanning the *x*-*y* plane. For the actual sensors (see Figure 1) the transmit and receive antennas are vertically oriented, thus we only simulate the vertical polarization.

Additionally, we assume here that the radiation efficiency, η is one so that the terms directivity and gain are used interchangeably.

1. Planar Array Size versus Half Power Beam Width

Given a random planar array of half-wave dipoles parallel to the *z*-axis, we wish to find the half power beam width in the direction the array has been steered. The broadside half power beam width of an array with wavelength λ and length L is: $HPBW = 50.8° \frac{\lambda}{L}$. This also applies to a random linear array of length L [19]. We now consider the uniform random planar array, steered to θ_o = 90° and ϕ_o = 0°. For observation angles in the *x*-*y* plane, $\theta = 90°$. Here we are interested in observations angles near the main beam, i.e. where ϕ is small and $\cos\phi \cong 1$. Thus Equation (14) reduces to:

$$AF(\theta = 90^{\circ}, \phi) = \sum_{n=0}^{N-1} A_n e^{jkx_n(\cos\phi - 1)} e^{jky_n \sin\phi}$$

where x_n and y_n range from 0 to L and all A_n are equivalent. When $\phi = 0°$ this is then the broadside of the array where the gain is maximum. For observations where ϕ is small, the length L in the y dimension determines the HPBW. Because $\cos\phi$ changes much more slowly than $\sin\phi$, the contribution due to the x component is relatively small. Thus the x component of the element has a very minor influence on the HPBW in the broadside observation. Therefore the expression for the HPBW of a linear random array is approximately valid for this case of the random planar array. Because the distribution of elements within the $L \times L$ area is random, this makes the choice of coordinate system arbitrary. If the pattern is constrained to the *x*-*y* plane, the expression for HPBW is valid for an arbitrary steering angle in the *x*-*y* plane. For larger values of ϕ, the array factor is dominated by the random character of the elements as demonstrated in Figure (12). Having shown that the HPBW of an N element random array can be estimated using Equation (16), we now validate this through MATLAB simulation of the random planar array. In this simulation, we model the planar array by arranging elements with a uniformly distributed random spacing. In this simulation, we held the number of

elements constant at 20. We then varied the length of the sides of the array, in terms of wavelength and calculated the HPBW at $\theta = 90°$ and $\phi = 0°$ for the vertical polarization. The coordinate system is the same as in Figure 10. For each array size, we generated 50 realizations of the array of random elements and then averaged the HPBW values from each to form a single data point. At each array size, Equation (16) was evaluated and plotted as the theory. The results of the simulation are presented in Figure 14.

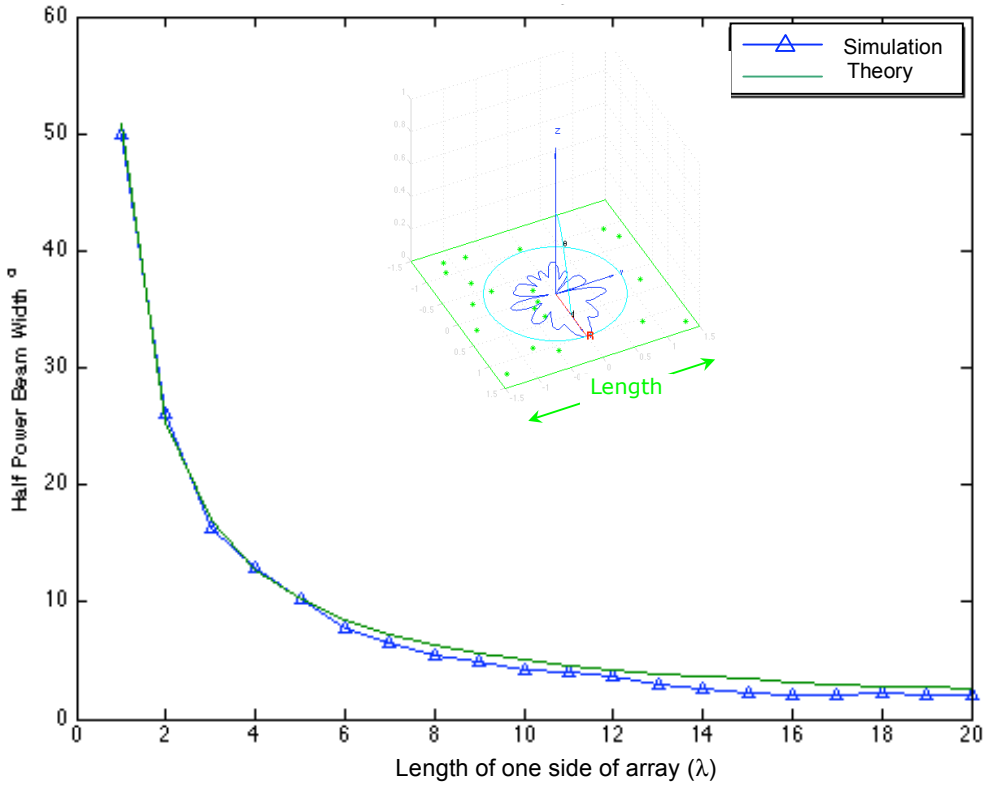


Figure 14. Broadside Half Power Beam Width for 20 Element Random Planar Array and Theoretical Half Power Beam Width Calculated for Increasing Array Size

We can see that as the array size increases, the HPBW becomes smaller. The results show good agreement with the theory, indeed the analytical expression seems to form an upper bound. From this result, we can use the array size to vary the beamwidth when the WSN is either searching for other WSNs where we want a relatively wide beam, or communicating directly with another WSN where we want a high gain narrow

beam. It should be noted that as the array size grows the spacing between elements will become large relative to a wavelength, especially when the array size is $20\lambda \times 20\lambda$.

2. Planar Array Maximum Gain versus Number of Elements and Average Side Lobe Level

In this section we examine how the maximum gain of a planar array is affected by increasing the number of elements for a fixed size array. In Section III.A.2, we observed that the gain of an N element random linear array approximates that of an N element linear (design) array; additionally, the average side lobe level is $1/N$ below the maximum gain and the peak side lobe level is no more than 10 dB above the average side lobe level. This simulation will validate these observations for the random planar array.

In Figure 15, we examine the maximum gain of the N element random planar array of half wave dipoles versus the number of elements. The physical size of the array is held constant at $5\lambda \times 5\lambda$. The maximum gain is calculated for the steering angle θ_o = 90° and ϕ_o = 0°, and is plotted in blue. Also plotted are the average side lobe level in red and peak side lobe level in cyan. The average side lobe level was computed by averaging all the peaks of the side lobes for each given number of elements. For each number of elements, 75 realizations of the random array were computed in order to form an average value and the datum plotted. The green line representing the theoretical average SLL is from Section III.A.2.c and is estimated as $1/N$ below the maximum gain. The purple line is the theoretical gain for a linear array of N dipoles and is $10\log_{10}(N)$ plus the gain for a half wavelength dipole of 2 dB [18]. The peak side lobe level in cyan is presented for reference. As discussed in Section II.A.2.d above, the peak SLL is less than 10 dB above the average SLL. The cyan line shows this and is clearly in agreement with theory. More exact, but significantly more complicated predictions for the peak SLL are presented in Chapter 8 of [19].

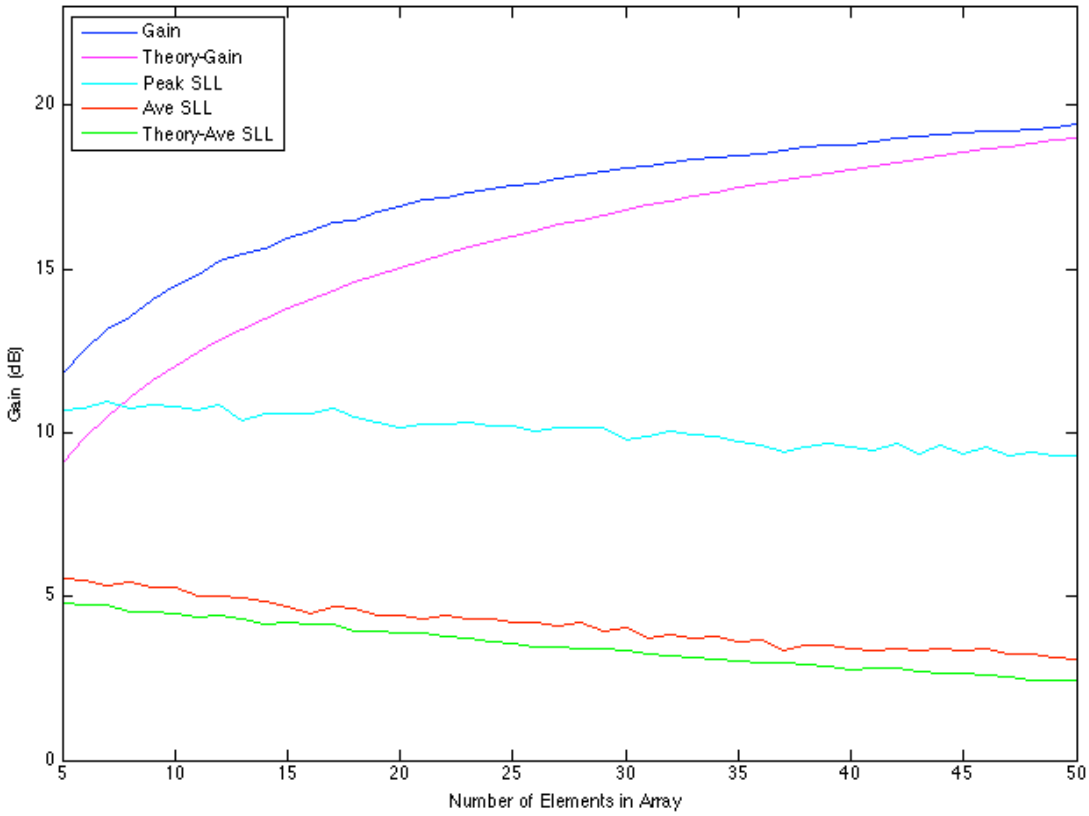


Figure 15. Maximum Gain, Average Side Lobe Level, and Peak Side Lobe Level as a function of the Number of Elements in a Random Planar Array of Size $5\lambda \times 5\lambda$. Theoretical Values Are Shown for the Maximum Gain and Average SLL.

In Figure 15 we see that the maximum gain is increasing with an increasing number of elements in the array and that as the number of elements increases, the average SLL is decreasing farther and farther below the maximum gain. The peak side lobe level is also decreasing as the number of elements increases. This is in agreement with theory.

Note that the maximum gain comes from the main lobe of the array, and is therefore unaffected by the random nature of the array, but rather is proportional to N [18]. This is in agreement with the plot. The difference in gain for 30 or more elements is less than or equal to 1 dB and the simulation and theoretical results converge as N increases. Overall, the theoretical and simulation curves are clearly in agreement.

From the theory of the random array, the average side lobe level should be less than the main lobe by a factor of 1/*N*. This is depicted by the red and green lines in Figure 15. The theoretical and simulation results again show excellent agreement. For example, at $N = 20$, the average SLL should be 13 dB below the peak. Here the peak is approximately 17.0 dB while the average is approximately 4.7, yielding a difference of 12.3 dB. At $N = 50$, the difference is 16.3 dB versus a calculated 17 dB.

These results show that we can vary the amount of gain for a fixed size array by varying the number of elements we choose to include. Thus, for a particular gain level required to form the proposed communications link, we may save battery power by only using the minimum required number of sensor nodes to transmit the message while simultaneously extending the overall lifetime of the WSN.

3. Planar Array: Gain versus Array Size for a Fixed Number of Elements

In a periodic array, gain increases with array size because elements are uniformly spaced. In a WSN, we may wish to achieve a particular gain level without increasing the number of elements. In this simulation, we demonstrate that we can thin a random planar array and maintain a constant gain. We fix the number of elements in the array at 20, then increase the size of the array from a $1\lambda \times 1\lambda$ to a $20\lambda \times 20\lambda$. At each array size, we compute the maximum gain, average SLL and peak SLL as was done in Figure 15.

Figure 16 shows the results of this simulation. We note that for a periodic array, the gain is actually due to an increase in the number of elements. Here we see that for a fixed number of elements (20), the gain shown in blue is relatively constant over a varying physical size. Since the number of elements is held constant and the physical size is increasing, we can see the effect of reducing the element density, or alternately, *thinning the array*. Also note that the average side lobe level shown in red is approximately 13 dB below the main lobe gain, which is given by Equation (15) and shown in green. As expected, the peak SLL does not exceed the average SLL by more than 10 dB. This is shown in cyan.

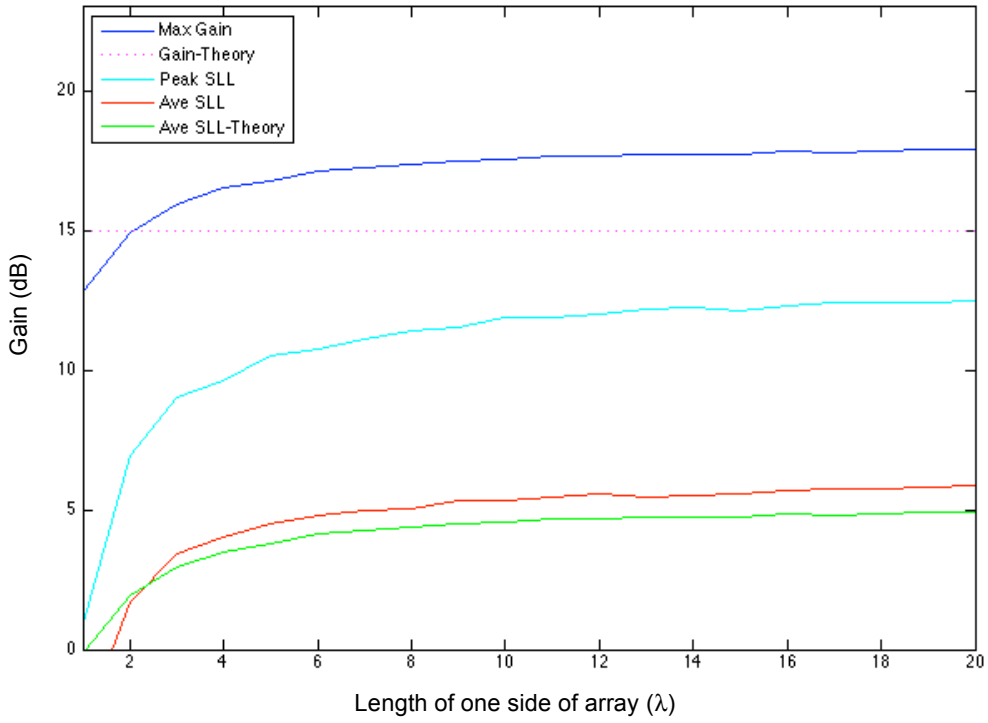


Figure 16. Maximum Gain, Average and Peak Side Lobe Levels as a Function of Random Planar Array Size for Fixed Number of Elements = 20. Theoretical Values Are Shown for the Maximum Gain and Average SLL.

As in Figure 15, the measured maximum gain is in agreement with theory; however, here it stabilizes approximately 2 dB above the theoretical level for a 20 element array of dipoles of 15 dB, shown in purple. The simulation appears to break down for array sizes less than 4 wavelengths, yet it quickly converges to a constant gain.

The independence of gain versus array size is an important result. For the WSN, we may require a particular gain level as determined by the number of elements *N* composing the array. From the results of Figure 16, we may choose *any N* elements as long as their distribution is still random within the bounded area [19].

4. Summary of Results

Figures 14, 15 and 16 demonstrate some important results that we can employ when building the proposed communications link between WSNs. The particular WSNs

are assumed to be fixed in location, but the inter-element spacing follows a uniform random distribution within a defined boundary. In order to link the WSNs, we wish to control the width, gain and direction of the array beam. We have shown in Equation (14) that we can steer the beam in an arbitrary direction. In Figures 14, 15 and 16 we have demonstrated control over the beamwidth and gain.

To achieve a given beamwidth, we choose a physical size, likely a sub-section of the WSN under consideration. The physical size controls the HPBW. Figure 14 and Equation (16) demonstrate this behavior. Within this physical area, we have control over the gain as demonstrated by Figure 15 by choosing the number of elements employed in the array.

Alternately, for a given required gain level, Figure 16 shows that we can thin the array (or reduce density) dramatically without significant loss of gain or the effects of grating lobes. It does not matter which particular sensor nodes within the WSN are used as long as, on average, they are randomly distributed within the physical area chosen. This is important as we may wish to distribute the burden of transmitting, and thus consuming energy among the nodes within the WSN.

To continue from Chapter II, in the first example using a pair of 50 element random planar arrays, with transmitters operating at full power (1mW each) and a margin of 15 dB, we determined the combined gain to be $(G_t + G_r) = 15 - N_t + 44$ dB. From Figure 15, we see that by choosing $N_t = 25$, we have a transmitter array gain G_t of approximately 17.5 dB. Therefore, we require a receiver array gain G_r of approximately 16.5 dB. This can be achieved with 17 receive elements. In the second link analysis where the transmitters are operating a minimum power, the required combined gain must satisfy $(G_t + G_r) = 15 - N_t + 68$ dB. Again, using Figure 15, we see that using 45 transmitter elements gives us a transmitter array gain G_t of 19 dB. The required receiver array gain G_r is approximately 19 dB. This can be achieved with approximately 45 receive elements.

Also consider a $20\lambda \times 20\lambda$ area with 240 elements contained within the boundary. Suppose that the gain requirement is such that we only need 15 elements. We

are still free to choose a beamwidth based upon the communication link needs. Suppose that a beamwidth of 10° is required, which by Equation (16) implies a $5\lambda \times 5\lambda$ sized array. Clearly, the WSN may subdivide itself into $5\lambda \times 5\lambda$ segments that contain at least 15 elements. The sub-sections could then time share the burden of transmitting. Vincent in [5], provides methods for power conservation and burden sharing of the WSN.

Armed with these results for determining the beamwidth and gain, we are now in a position to apply them to simulate a WSN searching for other WSNs.

C. SEARCH METHOD

In order for one WSN to find another, a method must be employed to systematically search the horizon. In this thesis we have chosen to employ a "lighthouse" approach in that we form a narrow beam and transmit in a given direction, then steer the beam across the horizon in search of an adjacent WSN. This section details some aspects of this method, such as beam steering, allowing for response time, power consumption, range capability of the array, calculation of the relative range and bearing between WSNs. The content of the initiating message is also discussed here.

1. Beam Steering

In this chapter, we have established that a random planar array can be used to form a beam, and this beam can be steered to a given direction. Equations (13) and (14) allow us to steer the array and the beamwidth is given by Equation (16). We know that beamwidth is related to the size of the chosen array (the longer the baseline, the thinner our beamwidth and vice versa), and that gain is related to the number of elements used. For example, to scan the horizon we may choose a half power beamwidth of 30°, transmit a signal in search of another WSN, then steer the beam ahead 30° and transmit again, etc., until we have moved around the entire horizon.

2. Look – Wait – Rotate Beam

This search method is such that the searching WSN will pick a start direction, transmit a signal in that direction and wait for a response. If no response is received, then the searching WSN will advance the beam and "look" in the next direction.

Since each node in the WSN acts as one element of the smart antenna, it has a particular phase or weight associated with it to steer the beam in each look direction. The weights can be stored in a table and sequenced through in order to advance the scan direction. These steering weights are the β_x and β_y of Equation (13). Since the geometry of the WSN is fixed, the steering weights associated with each sensor node (or transmit element) do not change and therefore do not need to be continuously recalculated. All nodes within the WSN transmit with the same amplitude. As noted in Section III.A.2.d on random array theory, tapering the current amplitude weights A_n of Equation (12) with respect to location in the array has little affect on the performance of the array. Since all elements transmit with the same amplitude, no complex current amplitude weighting scheme need be designed, simplifying the beam steering.

The technique of rotating the beam and only looking in one direction at a time was chosen to avoid having several WSNs respond at once as might occur if a single omnidirectional beacon were used. If a single beacon signal were sent omnidirectionally, the searching WSN would need to determine the direction of arrival from each of the responding signals, possibly at the same time.

3. Power Conservation

Power conservation is not a major focus in this thesis; however, the results of this work are meant to complement other research on energy conservation in WSNs [5], [9] , [21].

When initially searching for another WSN, the array does not know the distance to the other array, and therefore must expect the longest distance possible. This implies using all elements available within a subset of a given area to form the initial array, to maximize gain. Since we initially want a relatively wide beam, for fewer total beams in

the search and thus fewer transmissions, these constraints limit the physical size of the employed sensor transmitters. Thus a tradeoff will be required to optimize power consumed to find another WSN and establish the link.

4. Range and Bearing

It is assumed that one goal of the WSN is to extend its lifetime as long as feasible. Since the sensor nodes are battery powered, and using the radio transmitter consumes a relatively large amount of power [5], we wish to minimize the number of transmitters used in forming the communications link. We wish to only use the minimum number of transmitters required for a given range. Since a higher number of nodes equates to a higher transmission gain G_t, from the Friis equation, we can determine the amount of gain required for a given distance d [12]:

$$P_r = \left(\frac{\lambda}{4\pi d} \right)^2 P_t G_t G_r \tag{17}$$

where P_r is the received power, G_r is the receiver gain, λ is the wavelength and d is the range. From Equation (17), we see that a higher transmitter gain produces a higher received power P_r, thus for a given receiver gain a pair of WSNs will be able to communicate at a longer range if more nodes are used.

It is assumed that within the initial exchange of information between the searching WSN and receiving WSN are the respective latitude and longitudes. From this pair of lat/long coordinates, each WSN must calculate a bearing and distance to the other. For a given pair of latitude and longitude coordinates $(lat_1, long_1)$ and $(lat_2, long_2)$, the distance between them is given by the Haversine formulas [22]. Here, the WSNs are assumed to be on the order of a kilometer apart and the flat Earth approximation to the Haversine formulas is applicable [22]. The separation distance D (in km) and bearing Φ (in radians) are calculated using Equations (18) and (19):

$$D = R\sqrt{a^2 + b^2 - 2ab\cos(\Delta long)} \tag{18}$$

$$\Phi = \tan^{-1}\left(\frac{\sin(\Delta long)\cos(lat_2)}{\cos(lat_1)\sin(lat_2) - \sin(lat_1)\cos(lat_2)\cos(\Delta long)} \right) \tag{19}$$

where $a = \frac{\pi}{2} - lat_1$, $b = \frac{\pi}{2} - lat_2$ and $\Delta long = long_2 - long_1$. A simple java-script of the Haversine formulas for the flat Earth assumption is available at [23]. Once the range and bearing are known, the WSNs can calculate the required gain to establish the link. This gain calculation was the subject of Chapter II. For the sensor nodes, presumably deployed on the ground, the antenna height h_{ant} is approximately 0.1 m. The radio horizon for this node is $d = 4.12\sqrt{h_{ant}}$ [24] or 1.3 km. Thus the flat Earth approximation used in Equations (18) and (19) for this short distance seems applicable.

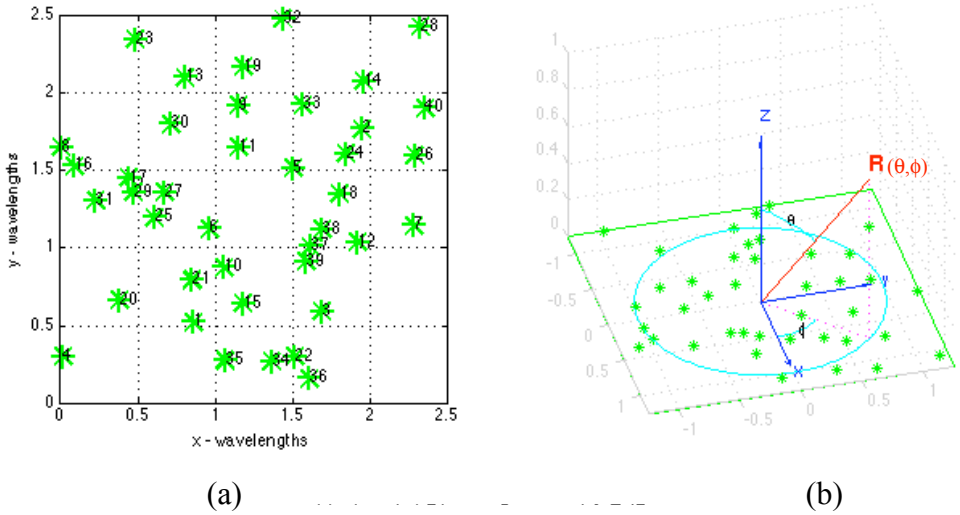
We now demonstrate a specific simulation of a WSN forming a beam and scanning it across the horizon.

D. MATLAB SIMULATION RESULTS – SCANNING BEAM

We now wish to combine the concepts of this chapter on random planar arrays with the idea of a searching WSN that transmits a beam in a rotating fashion across the horizon. We begin by choosing a half power beamwidth of approximately 20 degrees, from Equation (16), and set the array size at $2.5\lambda \times 2.5\lambda$. In this example we do not have a specific gain requirement to meet, so we pick 40 elements to obtain a gain of 16 dB as a convenient number. To steer the beam, we change the phase of each element in accordance with Equation (13).

Figure 17a shows the realization of a random array of 40 half wave dipole elements distributed over a $2.5\lambda \times 2.5\lambda$ area, and in Figure 17b the coordinate system is presented for reference. The minimum spacing between elements is set to 0.1λ, but would need to be increased to realize this example with the MICAz mote. The 0.1λ spacing found in some parts of this realization is much denser than a traditional planar array of the same area. As before, $R(\theta,\phi)$ is used to visualize the coordinate system by representing an arbitrary angle. The cyan circle depicts the locus of observation points, beginning at $\theta = 90°$ and with $\phi = 0°$ on the x-axis. Figure 18 shows the gain pattern of

the array steered to θ_o=90°, ϕ_o = 0°. Figure 19 shows polar plots of the gain patterns for the first six beams to envision the scanning performed by the array.



(a) (b)

Figure 17. Searching WSN:
(a) Realization of a 40 Element, $2.5\lambda \times 2.5\lambda$ Random Planar Array
(b) Coordinate System with Scan Plane at $\theta = 90°$ Shown in Cyan

In Figure 18, we see the main lobe with a peak gain of 16.7 dB, comparable to the theoretical gain of 16 dB for a 40 element array. A random shape characterizes the side lobes, with the expected pedestal nature shown in Figure 12. The average side lobe level for all scan angles is at 17.1 dB below the main lobe. The half power beam width in Figure 18 is 22°, in close agreement with the expected HPBW of 50.8°/2.5 = 20°.

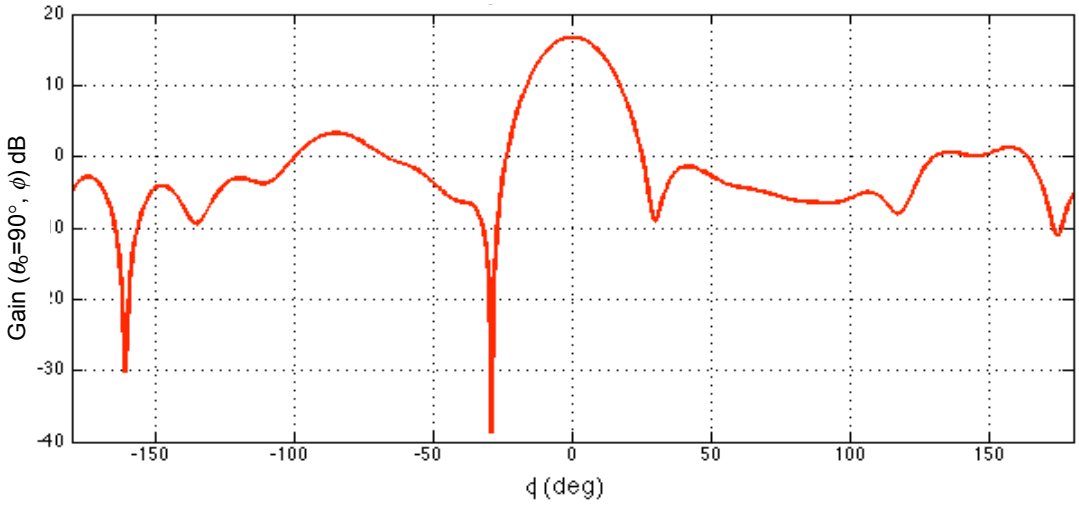
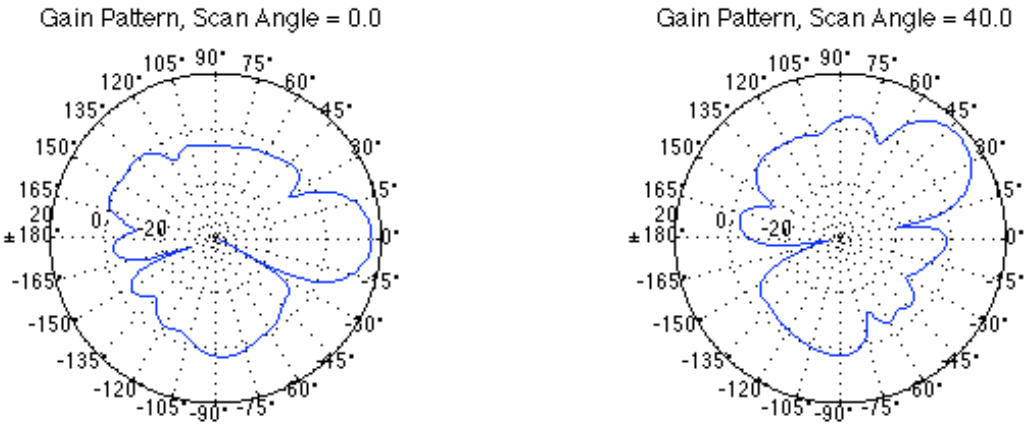


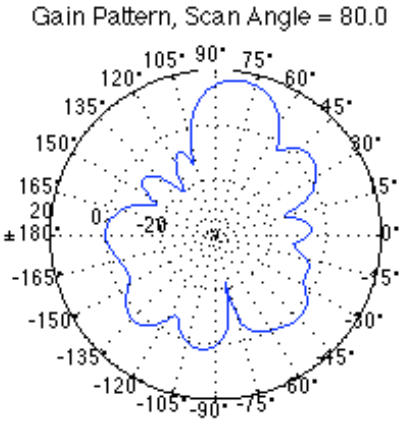
Figure 18. H-plane Gain Pattern for a 40 Element Random Planar Diploe Array with Beam Steering Angle θ_o=90°, ϕ_o = 0°

Figures 19a-f show the gain pattern for six consecutive beams in a polar plot format. Figure 19a is the same data as Figure 18 repeated in a polar format.

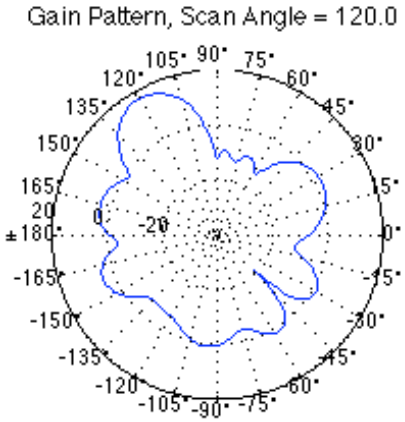


(a)

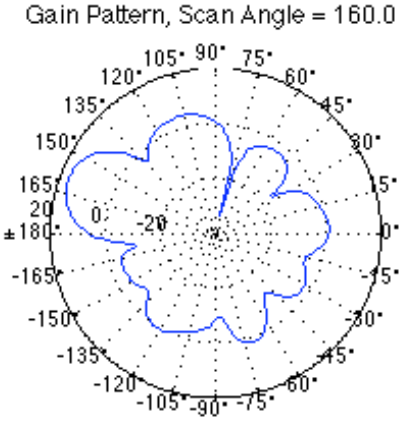
(b)



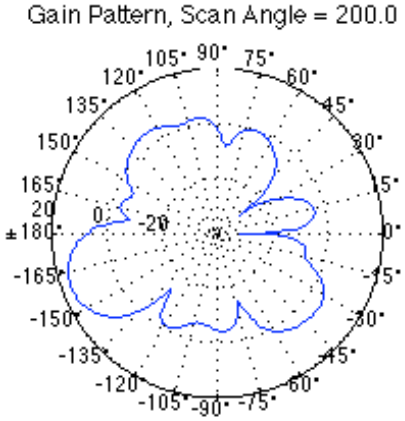
(c)



(d)



(e)



(f)

Figure 19. H-plane Patterns for a Random Planar Array of 40 Dipoles as Beam is Scanned From 0 to 200° in 40° Steps

We see that as the beam is steered in each direction, it retains approximately the same gain and average side lobe levels on each plot.

It should be noted that in this example we used a 20° beamwidth to reduce the number of beams required to scan the entire horizon, and thus the total power to complete one sweep. However, to obtain this beamwidth, the array size is limited to $2.5\lambda \times 2.5\lambda$, which in turn limits the total number of elements that can physically fit within that area. A reduced number of elements results in a lower peak gain. Thus a tradeoff is required between the total number of elements (peak gain) and the beamwidth (size of the array and total energy consumed to scan the horizon).

These plots demonstrate that the beam of a WSN composed of uniformly distributed, randomly placed sensor nodes can be used to form a beam and rotate that beam across the horizon. The characteristics of the antenna beam are consistent with the theory presented in this chapter on random antenna arrays. This completes the discussion on the transmitting or searching WSN. As the beam from the transmitting array scans the horizon, it is assumed to illuminate a receiving WSN, which must recognize and reply to this signal.

In this chapter, we presented results demonstrating control over the half power beam width, and gain and direction of the radiation pattern for a random planar array. We demonstrated that we could select a wide beam for the search pattern and simultaneously select the gain. This beam was steered in a lighthouse fashion simulating the search for and hopefully illumination of an adjacent WSN.

We now turn our attention to the receiving array.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. DIRECTION OF ARRIVAL

The approach of this chapter is essentially from the viewpoint of the receiving array. When the receiving WSN detects a signal from the searching WSN, it needs to know the relative geometries of the two WSNS in order to form its array and then send a reply. The location information of the sending WSN is contained within the message signal transmitted, but the direction of arrival (DOA) must be calculated. This chapter establishes the need to blindly determine the DOA of the received signal and presents a short survey of techniques available in the literature. A spread spectrum space division multiple access technique is introduced and demonstrated first with a linear array, and then applied to the random planar array. A simulation of the message extraction from a received signal is presented. The strengths of this technique as well as the importance of the spread spectrum pseudorandom noise codes are explained.

A. DETERMINATION OF THE RELATIVE GEOMETRIES

Here we consider two of the WSNs depicted in Figure 3, WSNs "A" and "B." We consider that WSN A has sensed a phenomenon and needs to pass this data to the next WSN, say WSN B. In Figure 20, WSN B has been illuminated by the searching WSN A and now needs to determine the DOA of the signal. The receiving WSN B will form a random planar array using its sensor nodes to analyze the incoming signal and determine the DOA.

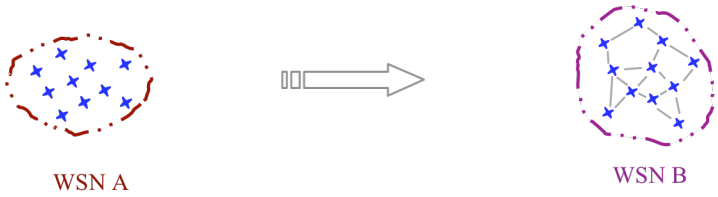


Figure 20. Signal From WSN A Illuminating WSN B

The receiving WSN will then use this calculated DOA and the location information encoded within the transmitted signal to determine the relative geometries and form the optimal return beam to the searching WSN.

1. Link Initiating Signal

In Figure 20, the receiving array has no prior knowledge of the searching WSN from which the initiating signal is coming. It will have to determine the direction of arrival by processing the signals received at each of the sensor nodes. Section B of this chapter discusses this in detail.

Some sensor nodes have the capability to operate in a power saving or "sleeping" mode. This thesis does not address the sleeping or transition of modes for each of the sensor nodes. Instead, we assume that either the search beacon contains a "wake up" signal or some other mechanism has allowed all sensors in the receiving array to be turned on for receiving signals.

2. Return Signal

We assume that in addition to the location of the sending array, the message in the search signal contains the "name" or "address" of the searching WSN, e.g., its MAC address and its lat/long coordinates. The receiving WSN is then in a position to solve its half of the link equation and form a return signal. For example it may have determined that WSN A at bearing 040° and range 500 m has requested a link. When it forms its reply or return message to the originator, it must include its lat/long coordinates as part of the initial handshake. The first task, however, is to determine the DOA of the query.

B. DOA DETERMINATION

Godara [10] provides a summary and comparison of various direction of arrival methods including spectral estimation, minimum variance distortionless response estimator, maximum likelihood, maximum entropy, least mean squares, conjugate gradient, MUltiple SIgnal Classification (MUSIC) and Estimation of Signal Parameters

via Rotational Invariance Technique (ESPRIT). MUSIC and ESPRIT are also known as eigenstructure or subspace methods and can be computationally intensive [10]. ESPRIT and most versions of MUSIC require periodic arrays [20], thus making them unsuitable for the WSN in our problem. Time difference of arrival (TDOA) is another technique used in many applications [9]. The direction of arrival problem is also known as angle of arrival, spectral estimation, or bearing estimation.

1. The Requirement for a Blind DOA Algorithm

From the perspective of the receiving WSN, when a signal is received from the searching WSN, its DOA is unknown. The receiving WSN must, therefore, determine the direction from which the searching WSN is transmitting without the aid of a reference signal as is used by some DOA methods. DOA determination methods that do not use prior knowledge of signal direction or some sort of reference signal to form an initial estimate for the DOA are termed blind DOA methods [10].

A new method based upon spread spectrum techniques for space division multiple access (SDMA) applications that does not involve iterative matrix solutions, reference beams or specific array geometry has been presented and described in a patent by Elam [25]. Because this method does not possess the computational intensity of the above methods, is suitable for random arrays and has the capability for blind DOA determination, we chose this method for the receiving array. The rest of this chapter discusses and applies the SDMA technique to our problem.

C. SPACE DIVISION MULTIPLE ACCESS RECEIVER

The direct sequence spread spectrum (DSSS) technique proposed by Elam [25] and as described by Gross [20] is summarized in this section. Analogous to the code division multiple access (CDMA) receiver, this space division multiple access receiver (SDMA) spatially separates received signals using DSSS techniques. The method involves chipping the phase of the received signals at each array element with individual spreading sequences. In computer memory identically chipped *virtual signals* are

generated from a virtual receiver array for I expected directions of arrival. The received signals are then summed and a quadrature correlation is performed between the received signal and each of the I *expected* DOA virtual signals. Correlation values (R_i) exceeding a threshold are identified as signals received from an expected DOA_i. From the phase of the correlation R_i, the message signal information may be extracted. Interfering signals are not well correlated and thus minimized [20]. This is similar in concept to a matched filter bank in a code division multiplexing receiver.

The time sample of the received signals is of length T. With this time sample, M pseudorandom chips are applied to the received signal at each receive element. Each receive element has a separate pseudorandom noise (pn) spreading code. The spreading sequences can be thought of as pseudorandom array weights. If viewed in the context of antenna array weights, during the time period T, the array has been "steered" by M sets of pseudo-random weights. Over all M chips, the average pattern approaches that of an omni-directional array [20]. Viewed another way, the "look direction" beams are created through the correlation with the virtual beams. Thus, steering the "look" direction of the physical array is done through the software that creates the virtual array expected DOAs.

We now discuss the receiver in greater detail. Figure 21 illustrates the receiver (after [20,25]). In the following sections we examine the three main components of the receiver: the actual array, the virtual array and the correlation and phase detection. We then describe the method for extracting the message in the received signal.

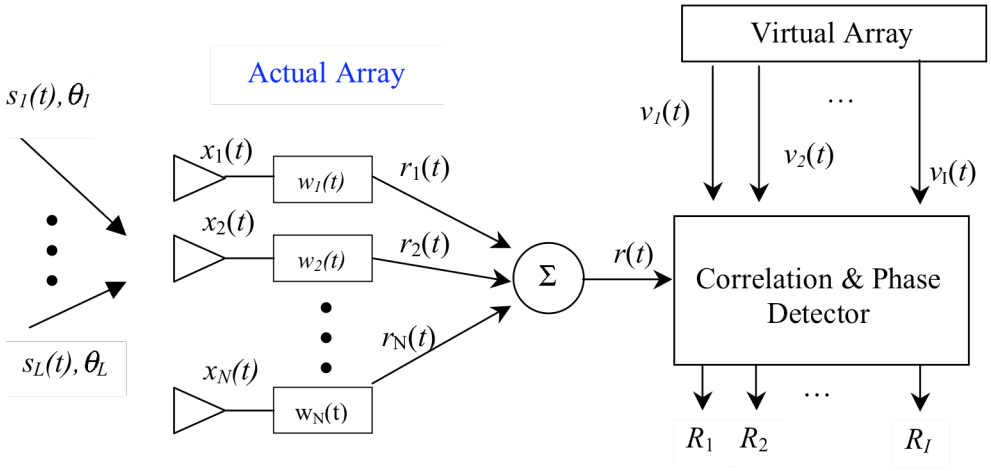


Figure 21. SDMA Receiver (after Gross and Elam)

1. The Actual Array

The actual array represents the physical receive antenna elements in the receiving WSN with a given geometry and L signals arriving at the array. The signals are assumed to have been phase shift keyed (PSK). The L phase modulated incident signals $s_l(t)$ arrive at the receiver and form the received signal $x_n(t)$ at each of the N antenna elements. The N elements need not be uniformly spaced, but can be arbitrarily located. Indeed, randomly placed elements, in a two or three-dimensional array helps the performance by producing independent phasing at each element for each expected angle of arrival. Each received signal $x_n(t)$ is then spread or chipped with a pseudorandom noise (pn) code associated with that antenna element, producing the spread signal $r_n(t)$, and finally the summed signal $r(t)$. The signal is sampled over a time period of t to $t+T$. The received signal at each array element can be expressed as:

$$x_n(t) = \sum_{l=1}^{L} a_{n,l}\, s_l(t) \tag{20}$$

where for a uniform linear array with k as the wavenumber and d the distance between elements, the array factor contribution $a_{n,l}$ for element n from signal l is given by:

$$a_{n,l} = e^{jkd(n-1)\sin\theta_l} \tag{21}$$

and the baseband phase modulated received signal of interest $s_l(t)$ is given by:

$$s_l(t) = e^{j\,\zeta_l(t)} \tag{22}$$

From Equation (20), we can express the received signal at each of the receive elements as an $N \times 1$ column vector:

$$\vec{x}(t) = \begin{bmatrix} x_1(t) & x_2(t) & \dots & x_N(t) \end{bmatrix}^T$$

For N elements and L signals, the received signal can be expressed in matrix form as:

$$\vec{x}(t) = \vec{A}\,\vec{s}(t) \tag{23}$$

where $\vec{A}$ is the $N \times L$ matrix of the steering vector elements, and $\vec{s}(t)$ is an $N \times 1$ column vector of the incident signals $s_l(t)$. The matrix $\vec{A}$ can be compactly expressed as a $1 \times L$ row vector of the steering vectors $\vec{a}_1^{\,r}(\theta_1)$:

$$\vec{x}(t) = \begin{bmatrix} \vec{a}_1^r(\theta_1) & \vec{a}_2^r(\theta_2) & \cdots & \vec{a}_L^r(\theta_L) \end{bmatrix} \begin{bmatrix} e^{\zeta_1(t)} \\ e^{\zeta_2(t)} \\ \vdots \\ e^{\zeta_L(t)} \end{bmatrix} \tag{24}$$

or $\vec{A}$ can be expressed showing the expanded steering vectors in order to see the individual contribution of the n^{th} element and l^{th} signal:

$$\vec{x}(t) = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ e^{jkd\sin\theta_1} & e^{jkd\sin\theta_2} & \cdots & e^{jkd\sin\theta_L} \\ \vdots & & \ddots & \vdots \\ e^{jkd(N-1)\sin\theta_1} & & & e^{jkd(N-1)\sin\theta_L} \end{bmatrix} \begin{bmatrix} e^{\zeta_1(t)} \\ e^{\zeta_2(t)} \\ \vdots \\ e^{\zeta_L(t)} \end{bmatrix} \tag{25}$$

For example, for $L = 1$, $x(t)$ is given as:

$$\vec{x}(t) = \begin{bmatrix} e^{j\zeta_1(t)} \\ e^{j\zeta_1(t)} e^{jkd\sin\theta_1} \\ \vdots \\ e^{j\zeta_1(t)} e^{jkd(N-1)\sin\theta_1} \end{bmatrix} \tag{26}$$

The pn chipped signal $r(t)$ is described in matrix form as:

$$r(t) = \vec{w}(t)\vec{x}(t) \tag{27}$$

where

$$\vec{w}(t) = \begin{bmatrix} w_0(t) & w_1(t) & \cdots & w_{N-1}(t) \end{bmatrix} \tag{28}$$

is the $1 \times N$ row vector of the spreading codes. These spreading codes $w_n(t)$ will be used to *phase modulate* the array outputs $x_n(t)$. Each vector $w_n(t)$ contains M chips of length τ, chosen such that the length of the vector is $M\tau = T$, the sample time. The number of chips M should be chosen such that it is greater than the Nyquist rate of the received baseband signal modulation [20]. This implies the phase modulation of the received baseband signal is nearly constant over T. The time-bandwidth product of the sample time T and the message signal bandwidth B should be $TB \leq 0.25$ [20].

In spread spectrum applications, there are many types of pseudorandom noise spreading codes available including maximal-length sequences or *m*-sequences, Gold codes [26], and Golay codes [27]. Walsh-Hadamard codes [27] and Welti codes [28] can

be used as orthogonal covering spreading codes. In our implementation we have chosen Walsh-Hadamard codes because of the simplicity of implementation.

The cross correlation of codes within the same family provides one measure of how well a spreading code works [12]. Codes that have a minimum cross-correlation will cause a minimum interference with other codes because signals spread with one code will not correlate with another code and be suppressed [26]. The orthogonality of the Walsh-Hadamard or Welti codes provides separation for each of the channels, here represented by receive elements which separate out the signals from each direction [20]. Welti codes have been reported to produce the best results in this application [29].

Figure 22 shows the effect of the spreading codes on the received signal at one of the antenna elements. Here we show the received signal phase $\zeta_n(t)$ at one of the antenna elements in blue, the Walsh-Hadamard spreading code w_n in red, and the real part of the chipped signal $r_n(t)$ in purple. The chipping process is repeated at each of the N receive elements, each element having its own individual Walsh-Hadamard code.

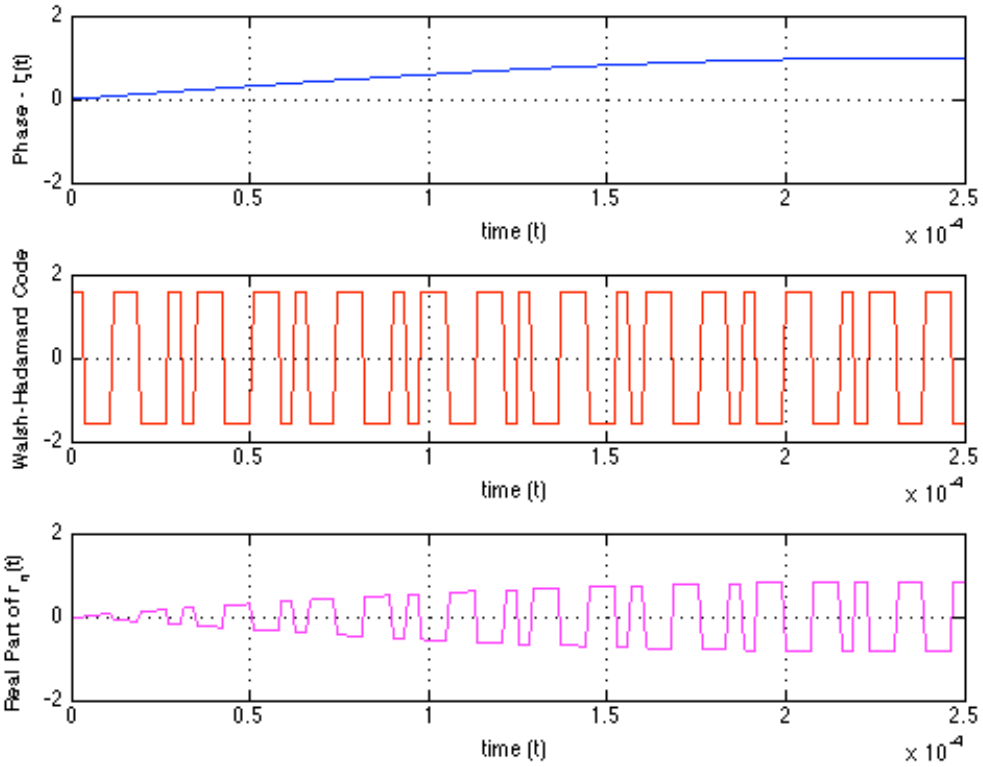


Figure 22. Received Signal Phase $\zeta(t)$, Walsh-Hadamard Spreading Code w_n, and the Real Part of the Chipped Signal $r_n(t)$

Key to this technique is the formation of the virtual array of signals that "look" in the expected directions of arrival. The generation of the virtual signals is similar to that of the actual array. We continue with the description of the SDMA receiver by examining the virtual array component.

2. The Virtual Array

Figure 23 graphically shows this component of the receiver. Within computer memory, a "virtual array" is configured according to the geometry of the physical array, containing the same number of elements. The virtual array is steered to an expected DOA by computing the array factor for that expected direction θ_i. The expected direction θ_I can also be thought of as the beam steering angle θ_o . The array factor for each of the N

elements is chipped with the *identical spreading codes* as the physical array. This is repeated and stored for *each* of the I expected directions of arrival θ_i. The I expected directions can be chosen to avoid known interfering directions, prior knowledge of the incoming signal, multipath [20,25], etc. The expected directions cannot be chosen any finer than physically allowed by the actual array.

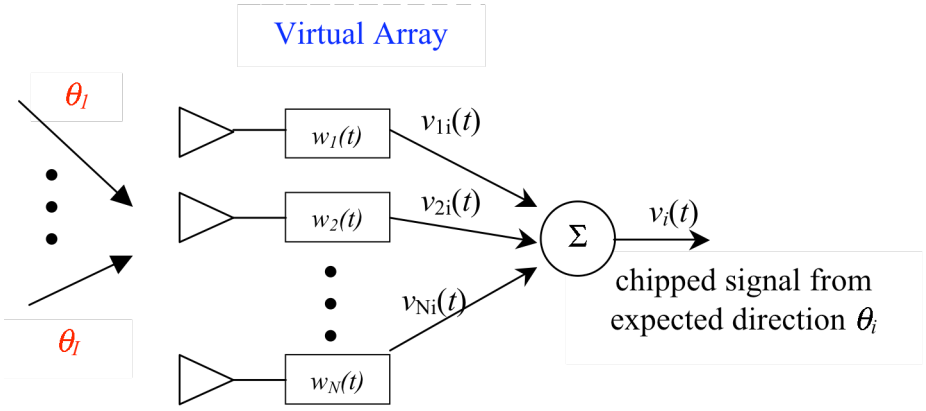


Figure 23. Expected Signal Generation by the Virtual Array in Memory

For each of the I expected directions, we can express the summed virtual signal as:

$$v_i(t) = \vec{w}(t)\ \vec{a}_i^{\,v}(\theta_i) \tag{29}$$

where the steering vector for each expected direction is given by the $N \times 1$ column vector:

$$\vec{a}_i^{\,v}(\theta_i) = \begin{bmatrix} 1 \\ e^{jkd\sin\theta_i} \\ \vdots \\ e^{jkd(N-1)\sin\theta_i} \end{bmatrix} \tag{30}$$

and the spreading codes are the same as before:

$$\vec{w}(t) = \begin{bmatrix} w_0(t) & w_1(t) & \cdots & w_{N-1}(t) \end{bmatrix}$$

Having formed both the received signal and the array of virtual signals, the receiver now performs a correlation computation to establish the DOA of signals incident to the receiver.

3. Signal and Phase Correlator

The *signal and phase correlator* correlates the actual received signal *r(t)* with each of the *I* virtual signals $v_i(t)$, over the sample time *T*. Figure 24 illustrates the operation.

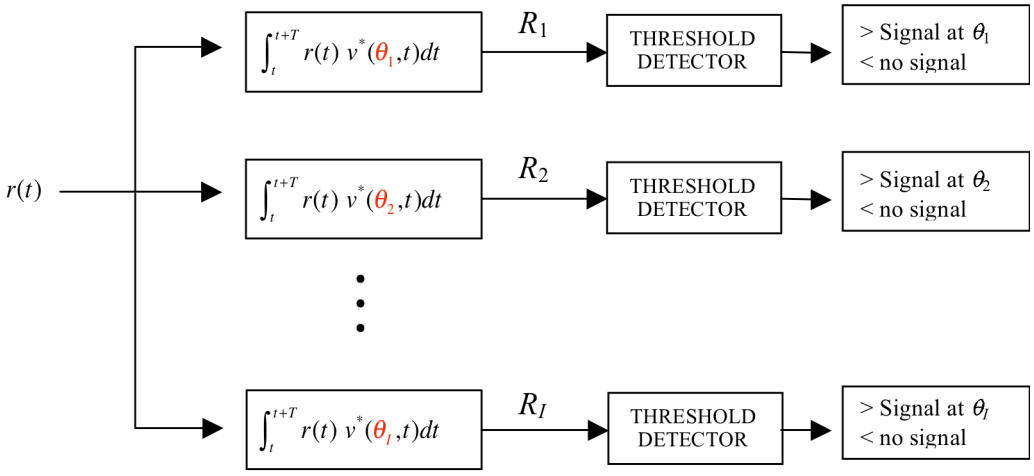


Figure 24. Signal and Phase Correlator

The correlation computation is expressed as:

$$R_i = \int_t^{t+T} r(t)\, v_i^*(t)\, dt \tag{31}$$

where R_i is the complex correlation value for direction θ_i. The magnitude of R_i will be plotted for each angle θ_i becoming the array factor for the signals present. The threshold detector depicted compares the magnitude of R_i against a predetermined value and declares a signal present or not at this angle. If a signal is present, the data is extracted as explained in the next subsection. This is similar to a matched filter or correlation detector operation [27].

The correlation values, when the pn codes for the actual and virtual signals match, become the array factor for the antenna array in the direction of the virtual signal. To see how this works, we examine the correlation in more detail. From Equations (22) and (27), the actual signal received at the antenna array *r(t)* with *N* array elements is expressed as:

$$r(t) = \sum_{n=1}^{N} e^{j(n-1)kd\sin\theta_0} e^{j\zeta(t)} e^{j\frac{\pi}{2}w_n(t)} \tag{32}$$

where the spreading codes w_n are modulating the phase of the received signal. The signal generated in memory for the i^{th} expected direction and modulated with the identical spreading codes is from Equation (29):

$$v_i(t) = \sum_{m=1}^{N} e^{j(m-1)kd\sin\theta_i} e^{j\frac{\pi}{2}w_m(t)} . \tag{33}$$

Substituting Equations (32) and (33) into (31) and noting that the signal has been sampled in time through the spreading process gives the expansion:

$$R_i = \int_t \sum_{n=1}^{N} e^{j(n-1)kd\sin\theta_o} e^{j\zeta(t)} e^{j\frac{\pi}{2}w_n(t)} \sum_{m=1}^{N} e^{-j(m-1)kd\sin\theta_i} e^{-j\frac{\pi}{2}w_m(t)} \tag{34}$$

Rearranging the element summation yields

$$R_i = \int_t \sum_{n=1}^{N}\sum_{m=1}^{N} e^{j(n-1)kd\sin\theta_o} e^{j\zeta(t)} e^{j\frac{\pi}{2}w_n(t)} e^{-j(m-1)kd\sin\theta_i} e^{-j\frac{\pi}{2}w_m(t)} \tag{35}$$

Since the array factor terms are independent of time, they can be moved outside the integral. Recall that we chose the time sample period T such that the phase $\zeta(t)$ of the signal *s(t)* was approximately constant over the sample period and can be approximated as its average $\bar{\zeta}$; thus, it can be moved outside the integral as well:

$$R_i = \sum_{n=1}^{N}\sum_{m=1}^{N} e^{j(n-1)kd\sin\theta_o} e^{-j(m-1)kd\sin\theta_i} e^{j\bar{\zeta}} \underbrace{\int_t e^{j\frac{\pi}{2}w_n(t)} e^{-j\frac{\pi}{2}w_m(t)}}_{\delta(n-m)=\begin{cases}1 \text{ if } n=m\\ 0 \text{ if } n\neq m\end{cases}} \tag{36}$$

The integral yields a delta function $\delta(n-m)$ because w_n and w_m are orthogonal [12]. Thus R_i reduces to:

$$R_i = \sum_{n=1}^{N} e^{j(n-1)kd(\sin\theta_o - \sin\theta_i)} e^{j\bar{\zeta}} \tag{37}$$

Equation (37) is equivalent to the array factor, here multiplied by the constant phase of the received signal. Consequently, the correlation can be expressed in terms of magnitude and phase as:

$$R_i = |R_i| e^{j\rho_i} \tag{38}$$

where $|R_i|$ is the magnitude of the complex valued correlation and ρ_i expresses the phase of the received signal. The magnitude of the correlation contains the DOA information and the phase term contains the message information.

4. Message Extraction

The phase of the signal *s(t)* can also be estimated, and the message data extracted using this SDMA method. Over the time sample period *T*, the phase of signal *s(t)* was assumed to be approximately constant. Thus, when a signal is detected from direction *i*, the phase and thus the message can be approximated as ρ_i. The correlation integral averages the received signal phase over *T* [20], therefore

$$\rho_i \approx \bar{\zeta} = \frac{1}{T}\int_t^{t+T} \zeta_i(t)dt \tag{39}$$

where $\bar{\zeta}$ is the average signal modulation during the time period *T* at θ_i. By calculating ρ_i at each angle where a signal has been determined present, we can simultaneously extract more than one message signal from the correlation data. The process is described in further detail in [20, 25].

5. The Importance of the Spreading Codes

One may ask what advantage do the spreading codes provide to the overall receiver? The answer, in short, is that the weight functions or the chipping sequences provide orthogonality of each received signal to all of the incorrect DOA virtual signals. To examine this, we consider an example with $N = 4$ elements; the signal is incident from direction θ_{inc} and w_n, *k* and *d* are defined as before. The received signal is then:

$$r(t) = e^{j\zeta_1(t)}\left[w_0(t)e^{j0} + w_1(t)e^{jkd\sin\theta_{inc}} + w_2(t)e^{jk2d\sin\theta_{inc}} + w_3(t)e^{jk3d\sin\theta_{inc}}\right]$$

Recalling that the time sample period *T* was chosen such that *s(t)* was approximately constant in phase, the leading exponential can be suppressed as a constant. If *r(t)* is not chipped, the time sample is reduced to a single sample point in time, $s(t_0)$, and the expression for *r(t)* becomes the summation:

$$r(t_o) = \sum_{n=1}^{N} d_n = \sum_{n=1}^{N} e^{j(n-1)kd\sin\theta_{inc}} \tag{40}$$

where d_n represents an array elements' contribution to the array factor.

For each of the I expected directions, the virtual signal is expressed as:

$$v_i(t_o) = \sum_{n=1}^{N} d_{ni} = \sum_{n=1}^{N} e^{j(n-1)kd\sin\theta_i} \tag{41}$$

where d_{ni} represents the virtual array elements' contribution to the array factor.

Using Equations (40) and (41) in the correlation computation yields:

$$R_i = r(t_o)v_i^*(t_o) = \sum_{n=1}^{N}\sum_{m=1}^{N} d_n d_{mi}^* = \sum_{n=1}^{N}\sum_{m=1}^{N} e^{j(n-1)kd\sin\theta_{inc}} e^{-j(m-1)kd\sin\theta_i} \tag{42}$$

This expression is the correlation sum that results when the spreading codes are not used. Now substituting:

$$\psi_{inc} = kd\sin\theta_{inc} \quad \text{and} \quad \psi_i = kd\sin\theta_i \tag{43}$$

into Equation (45) and expanding the sums, the correlation sum takes the form:

$$R_i = [e^{j0\psi_{inc}} + e^{j\psi_{inc}} + e^{j2\psi_{inc}} + e^{j3\psi_{inc}}][e^{-j0\psi_i} + e^{-j\psi_i} + e^{-j2\psi_i} + e^{-j3\psi_i}] \tag{44}$$

The correlation magnitude should be at a maximum when the virtual angle θ_i matches the incident signal angle θ_{inc}, i.e., $\psi_{inc} = \psi_i = kd\sin\theta_{inc}$ = a constant. After some algebraic manipulation, Equation (44) can be written as:

$$R_{\max} = 4 + \frac{3}{2}\cos(kd\sin\theta_{inc}) + \cos(2kd\sin\theta_{inc}) + \frac{1}{2}\cos(3kd\sin\theta_{inc}) \tag{45}$$

While our intent was that the correlation reached its maximum when the expected angle matched the actual incident angle, this clearly is not the case. In Equation (45), the cosine and sine terms will influence the value of R_i depending upon the particular value of θ for the incident signal. Thus evaluating the correlation without chipping codes will not give us the clear solution we are seeking. While the correlation between $r(t)$ and $v_i(t_o)$ should be large when θ_i matches the actual received direction of arrival θ_{inc}, there is no guarantee that for any other arbitrary angle θ_i, the correlation would not be large.

More generally, we can clearly see the advantage of the chipped sequence by comparing the correlation with and without the spreading sequence. In the non-spread approach, the correlation sum contains cross terms as seen in Equation (42) while the SDMA approach produces the array factor in which the cross terms are zero as given in

Equation (36). It is clear that the orthogonal spreading codes used in the SDMA technique are integral to the method.

6. Advantages of the SDMA Technique

The strength of this technique comes from the ability to receive several signals and "look" in almost all directions nearly simultaneously. Because the chipping of the phase of the received signals and the phase offset at each of the antenna elements, this essentially makes the set of possible antenna patterns independent during the sampling period. The orthogonality of spreading codes between antenna elements allows the actual signals to be extracted with the correct direction of arrival. The signal message is extracted from the phase of correlation sequence. Since the set of virtual signals is computed once and stored in memory, the computational cost is significantly reduced as compared to the traditional DOA methods, such as conjugate gradient and other iterative matrix solution methods. Additionally, the technique provides a processing gain of $\sqrt{N_{chips}}$ [25].

Having described the SDMA method we now demonstrate through simulation the effectiveness of the method as applied to a uniform linear array and a random planar array of antenna elements.

D. MATLAB SIMULATION RESULTS

In this section we demonstrate the results of the SDMA technique using MATLAB. First we apply the technique to an 11 element ULA and demonstrate the importance of choosing the optimum spreading codes. Having selected the Walsh-Hadamard codes we then show the affect of the number of chips used to spread the incoming signal. We then apply the technique to the random planar array and simulate solving for a DOA and then extracting a phase encoded message from the detected signal. This is a simulation of the WSN B of Figure 20 receiving a message from WSN A and then decoding the location message. In the message extraction we simulate a white Gaussian noise channel.

1. Uniform Linear Array

In this simulation we use the SDMA method to calculate the DOA of two simultaneous test signals. The ULA contains 11 *z*-direction oriented dipole elements spaced one-half wavelength apart and is used to detect test signals arriving from two different directions -30° and +45°. The array factor for this ULA is given simply as the sum of the array factors from each incident angle. For an *N* element ULA, the array factor for a single incident wave from direction θ_{inc} is [18]:

$$AF(\theta) = \frac{\sin\left(Nkd\left(\sin\theta - \sin\theta_{inc}\right)\right)}{N\sin\left(kd\left(\sin\theta - \sin\theta_{inc}\right)\right)}$$

and the complete array factor for two incident signals is the sum of both array factors:

$$AF_{Total}(\theta) = AF_{-30^\circ}(\theta) + AF_{+45^\circ}(\theta)$$

Again, the incident "look" angles can be thought of as the beam steering angles. Here, the incident beams are restricted to the *xz* plane. The geometry used in this simulation is shown in Figure 25.

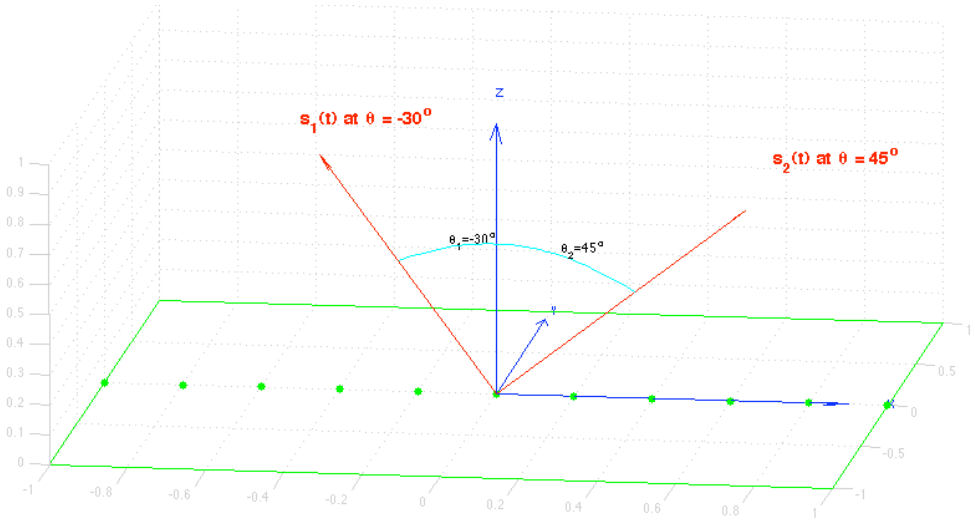


Figure 25. Geometry for ULA with Signals Incident from -30° and +45°

The choice of spreading codes has an influence on the quality of the DOA determination. In Figure 26, we compare the result of the correlation using two different

types of spreading codes, the Walsh-Hadamard codes and a pseudorandom noise code sequence. The correlation magnitudes and the array factor are plotted against the angle of incidence θ.

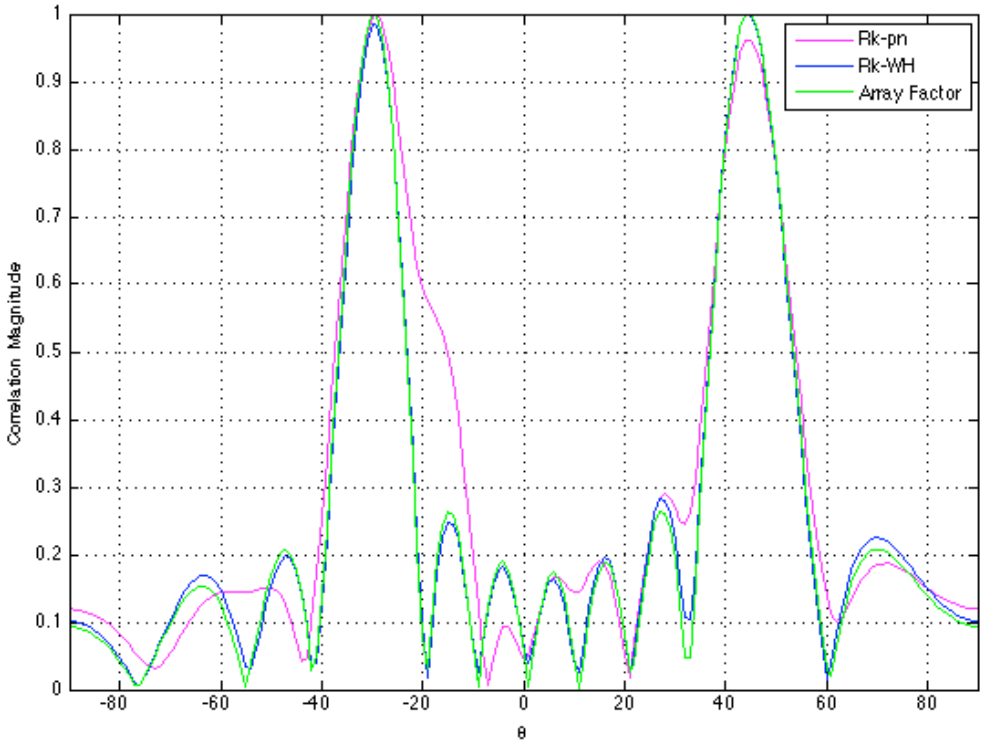


Figure 26. SDMA – Comparison of PN and Walsh Spreading Codes Using 32 Chips 11 Element ULA with Signals Incident from -30° and +45°.

The green line is the array factor. In this comparison, we used 32 chips for both the pn sequence and the Walsh-Hadamard codes and 4 samples of the signal per chip. A low number of chips was chosen in order to make the difference in spreading code choice more obvious. The blue line is the correlation magnitude calculated using the Walsh-Hadamard spreading codes and the magenta line is the correlation magnitude calculated using the pn sequence. In this figure, Walsh-Hadamard codes clearly work better than a pn sequence. The pn codes are adequate at determining the DOA of the signal, but the sidelobes do not match closely with those of the theoretical array factor. For a random

array, where the sidelobes do not decrease as strongly as the ULA, this suggests that the Walsh-Hadamard codes are a better choice.

We now examine the effect of the number of chips used. In Figure 27, we increased the number of chips used from 32 to 64 with the same number of samples per chip, array geometry and incident signals as in Figure 25. The array factor for the ULA is plotted in green and the magnitude of the correlation R_i is plotted in blue. We see from this result that the SDMA technique produces a correlation in excellent agreement with the theoretically produced array factor, clearly showing the DOA of the two incident signals at θ = -30° and +45° and accurately matching the mainlobe as well as all the sidelobes.

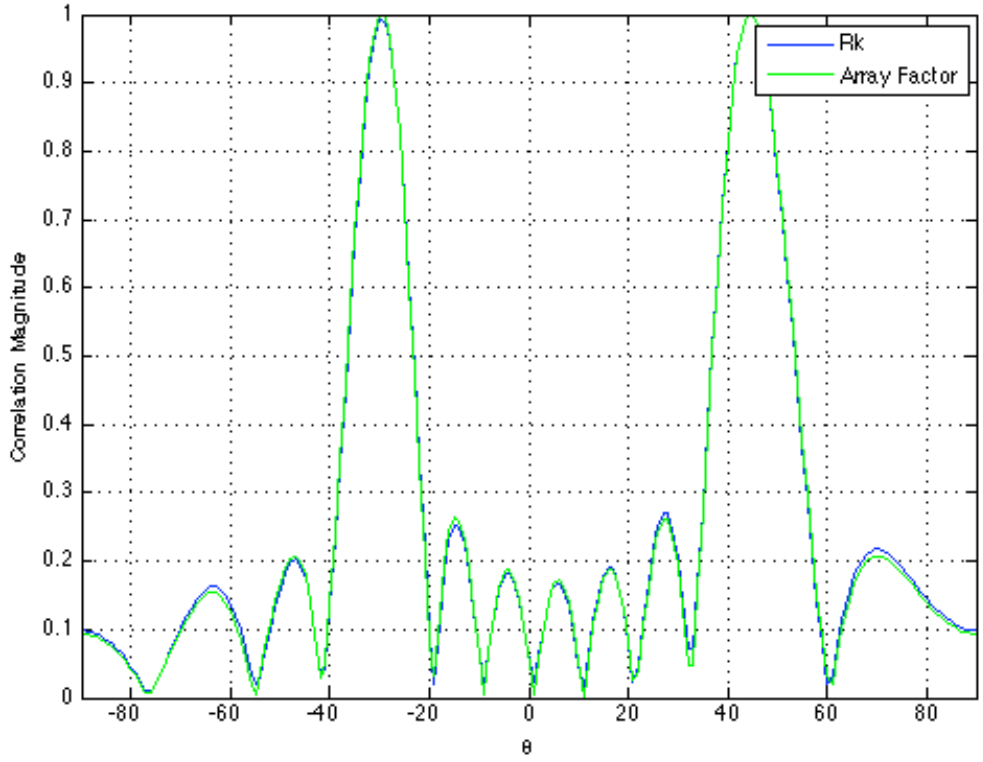


Figure 27. DOA Determination of Two Incident Signals Using SDMA on an 11 Element Array with One Half λ Spacing and 64 Chip Walsh-Hadamard Code

From Figures 26 and 27, we see that the choice of spreading code and the number of chips are important. In comparison to Figure 26, we see that 64 chips produce a much

closer match to the theoretical array factor than 32 chips when using Walsh-Hadamard codes on the same array. Additional comparisons (not shown) reveal that increasing the number of chips to 128 or greater had only marginal increase in the correlation magnitude as compared to theory. It appears that 64 chips are adequate for DOA determination.

Having demonstrated that the SDMA receiver performance agrees with theory for the uniformly spaced linear array, we now extend the technique to the problem of the planar array of randomly placed elements.

2. MATLAB Simulation Results – SDMA and Random Planar Array

In section A of this chapter, we discussed the need for the receiving array to determine both the DOA of the searching signal and the range to the searching WSN. Here we apply the SDMA technique to the random planar array of dipoles to determine the DOA of a test signal and compare the gain pattern from the SDMA technique to that of the array as calculated in Chapter III. Having found the DOA, we simulate the decoding of a sample message representing bits of a data stream from the searching WSN. The method for calculating the contribution of each array element to the correlation value for a random planar array is exactly analogous to that of Chapter III; instead of using a geometric expression for the location of a receive element we use the exact location as was done in Equation (14). We modify Equation (36) for the random planar array as follows:

$$R_i = \sum_{n=1}^{N}\sum_{m=1}^{N} e^{jk\psi_n(\theta_0,\phi_0)} e^{-jk\psi_m(\theta_i,\phi_i)} e^{j\zeta(t)} \int_{t}^{t+T} e^{j\frac{\pi}{2}w_n(t)} e^{-j\frac{\pi}{2}w_m(t)} dt \tag{46}$$

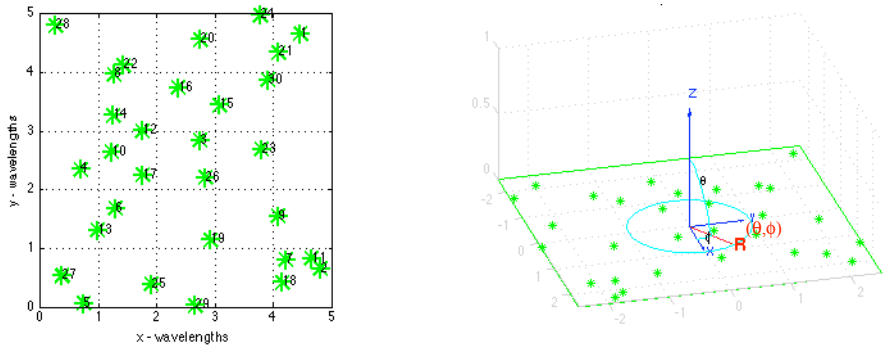
where the element phase $\psi_n(\theta_0,\phi_0)$ is now determined by its *x-y* coordinate and the steering angle (θ_0,ϕ_0) given by:

$$\psi_n(\theta_0,\phi_0) = (x_n \sin\theta\cos\phi - x_n \sin\theta_0\cos\phi_0) + (y_n \sin\theta\cos\phi - y_n \sin\theta_0\cos\phi_0)$$

a. DOA Determination

We now apply the SDMA technique to a planar array of randomly placed sensors. Figure 28a depicts an overhead view of the elements of a $5\lambda \times 5\lambda$ sensor WSN containing 30 elements, randomly distributed with the array scanning the *x-y* plane.

Figure 28b depicts the simulated test signal incident from $\theta = 90°$ and $\phi = 30°$. In this diagram $R(\theta,\phi)$ is shown to establish the coordinate system and indicates the direction of incidence of the test signal.



(a) (b)

Figure 28. Thirty Element Random Planar Array Realization and Coordinate System

With the geometry shown in Figure 28, the gain pattern of the random planar array was calculated using Equations (3) and (14) and compared to the SDMA gain pattern calculated using Equations (3) and (46). Equation (3) is the element pattern of the half wave dipole. In Figure 29, the green line shows the gain pattern of the random planar array with the incident test signal; it is the signal the SDMA technique must locate. The SDMA gain pattern is presented in blue and shows excellent agreement with the calculated gain pattern.

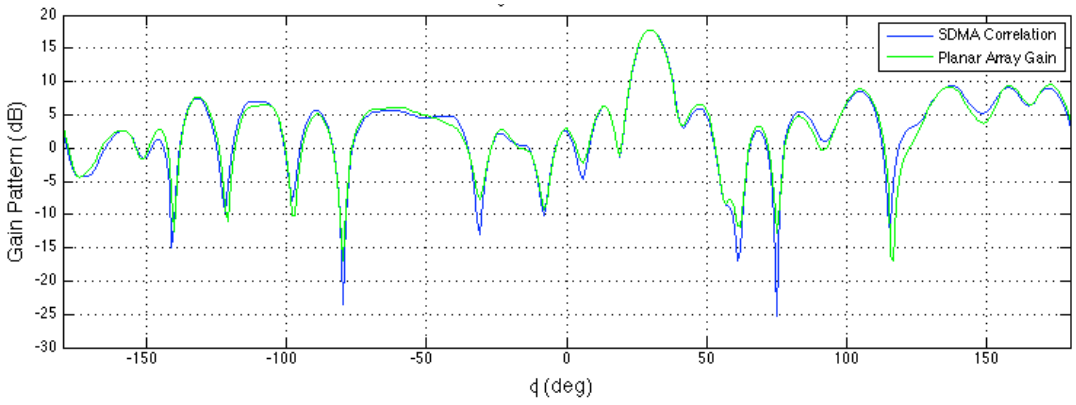


Figure 29. Gain Pattern of 30 Element Random Planar Array, Steered to 30° with SDMA Estimation of Incident Signal Pattern.

As shown in Chapter III, the random array produces a main lobe in the steered direction with the sidelobes displaying the characteristic random nature below a pedestal. The 30 elements produce a peak gain of 17.8 dB, the beamwidth is 8 degrees and the highest side lobe rises to 8.3 dB below the peak. These values are in agreement with the theoretical results presented in Chapter III. The SDMA pattern accurately shows the main lobe at 30° and all of the sidelobe peaks. The angle where the nulls of the gain pattern occur are accurately shown with the SDMA technique, but the absolute values do not exactly agree with the theoretical values.

These results validate the SDMA technique as a method for blindly determining the DOA when applied to the random planar array. It shows excellent agreement between the theoretical array patterns and those produced by MATLAB. This will allow the receiving WSN to almost instantaneously determine the DOA of the searching WSN.

Having determined the DOA of the searching WSN, the task of the receiving WSN is to extract the location information from the message signal sent by the searching WSN.

b. Message Extraction

One of the primary goals of the receiving array is to extract from the transmitted signal a message containing the location of the searching WSN. In order to demonstrate this, we will simulate the baseband message signal as a bit sequence and include it in the signal incident from 30°. The goal is to transmit an 8 bit sequence (0 1 1 0 1 0 1 1) mapped to (1 -1 -1 1 -1 1 -1 -1) as the sample message. The signal phase $\zeta(t)$ was generated with additive white Gaussian noise (AWGN) with a 20 dB signal to noise ratio to simulate channel interference. Here MATLAB measures the power in $\zeta(t)$ and adds AWGN at 20 dB. The signal to noise ratio is the ratio of the power in the signal ζ (t) to the power in the AWGN.

The results of this simulation are shown in Figure 30. We see in blue, the instantaneous baseband signal $\zeta(t)$ with AWGN. In green is the phase term $\rho_{i=30}$ of the correlation sequence R_i at the incidence angle $\theta = 30°$. The time sample of the received signals is of length T and shown in red. For simplicity, we send one bit per time period T, although this need not be the case [20,25]. We see that the signal bit sequence can be accurately extracted from the recovered estimate of the phase.

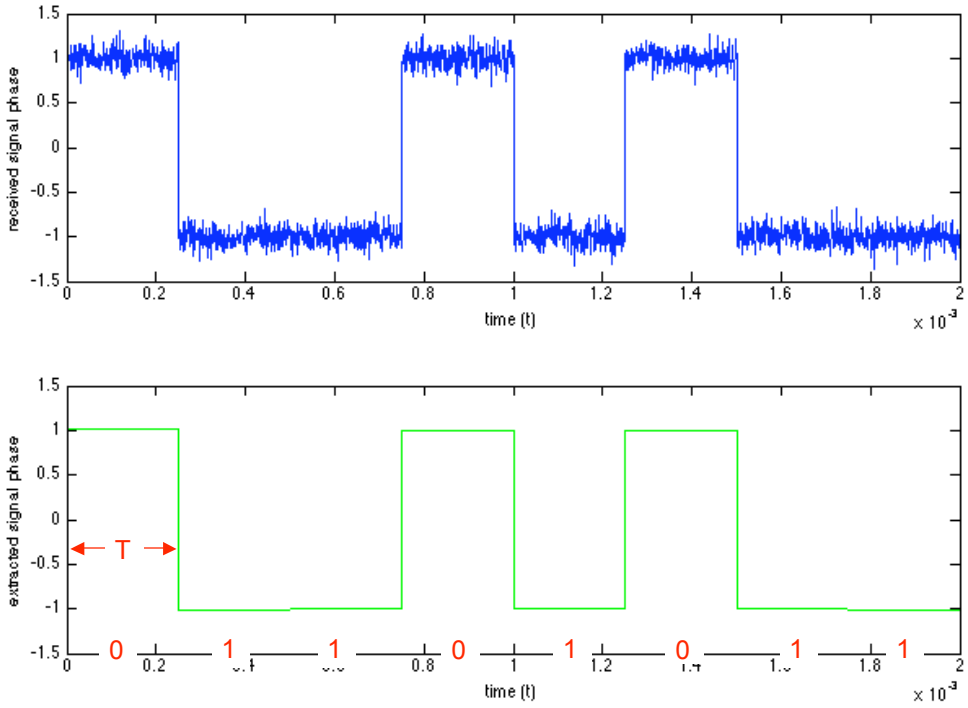


Figure 30. SDMA Message Extraction of Sequence [01101011] from Noisy Baseband Signal, SNR = 20 dB

In order to test the strength of the SDMA technique, we increased the noise level such that the SNR was now −5 dB. Figure 30 shows the received baseband test signal in blue encoding the same bit sequence at the higher noise level. The recovered signal is shown in green. Although the noise level appears very high, the SDMA technique is still able to extract the data in the incident signal. This suggests that the receiving WSN is capable of extracting the message bits under high noise conditions.

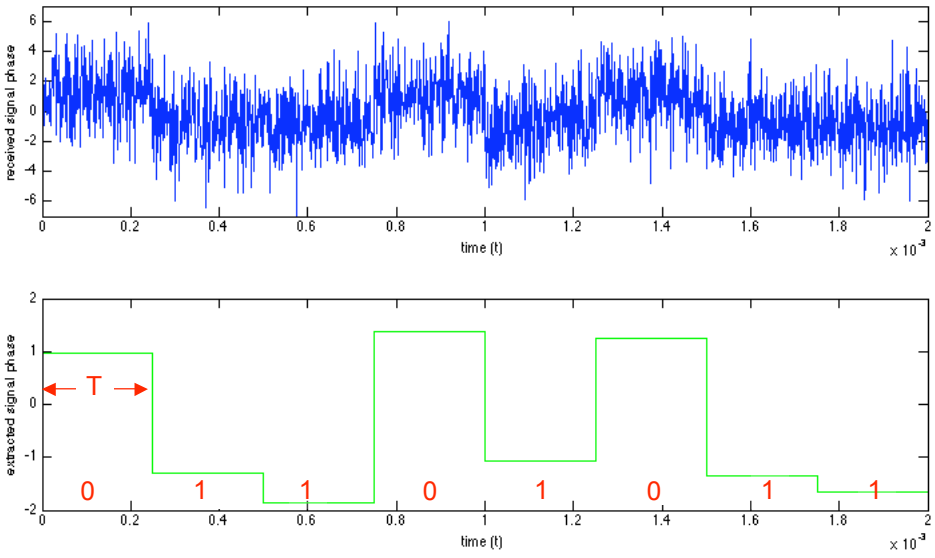


Figure 31. SDMA Message Extraction of Sequence [01101011] from Noisy Baseband Signal, SNR = –5 Db

The direction of arrival for a signal from a searching WSN is determined by the magnitude of the correlation sequence and the message itself is contained within the phase of the correlation. The bit stream is the message containing the "name" and "location" of the WSN searching for the adjacent WSN. Thus, the two key pieces of information required to establish the link, the bearing and range can now be determined. The latitude and longitude location information can be converted into a range using Equations (18) and (19).

The results shown here are preliminary to demonstrate the effectiveness of the proposed scheme. Further in depth simulation and comparison to gain a full understanding and scope of the proposed technique is required.

In this chapter, we established the need for the receiving WSN to blindly determine the DOA of an arriving signal. A new SDMA receiver was described and its effectiveness is demonstrated through the simulation of a random planar array. We showed that the method works better for Walsh-Hadamard spreading codes than for pn codes generated by a pseudorandom number generator and that a 64 chip sequence is

sufficient to accurately establish the DOA and extract the message. Using a random planar array, we validated the technique by determining the DOA of an incident test signal and then extracting a message data stream from this signal in a noisy channel at SNRs of 20 db and −5 dB. From the DOA and the location information within the arriving signal, the WSN has sufficient knowledge to respond to the query of the searching WSN.

Once the receiving array has knowledge it is being queried for a link, the task is then to focus its array towards the searching array, calculating the necessary parameters required for the link. This is done via the link budget analysis as was described in Chapter II, thus completing the link between two WSNs.

V. CONCLUSIONS

In this chapter, we present a summary of the work done, highlight the significant results and suggest topics for further research.

A. SUMMARY OF RESEARCH

In this thesis we explored methods for forming an Over The Horizon (OTH) communications link of Wireless Sensor Networks (WSNs) by enabling each WSN to act as a smart antenna array. Methods for establishing the OTH communications link via beam forming and direct sequence spread spectrum Space Division Multiple Access (SDMA) were presented and modeled using MATLAB. Methods for forming a search beam for a transmitting WSN and determining the direction of arrival of the received beam for a receiving WSN were presented.

We examined the searching WSN in which the individual sensor nodes operate in concert as a smart antenna array. The WSN then forms and transmits a search beam around the horizon in order to illuminate an adjacent WSN and establish a link. Each WSN was modeled as a uniformly distributed random array of sensors nodes. Gain and beamwidth of the random planar arrays were calculated and validated using MATLAB simulation and compared to theoretical results. Through MATLAB simulation, the half power beamwidth (HPBW) of a random planar array as a function of WSN size and the gain as a function of the number of elements for a fixed size array were studied. A 40 element random planar array, comparable to a 36 element periodic planar array of similar size, was simulated to demonstrate the search capability. .

We examined the proposed communications link from the perspective of the receiving array. This array needs to determine the direction of arrival (DOA) and location of the searching WSN in order to form an optimum link without a reference or pilot signal. A SDMA receiver capable of determining the DOA and location of the searching WSN was described and its effectiveness demonstrated through the simulation of a random planar array. Using a random planar array, we validated the SDMA method

by determining the DOA of an incident test signal and then extracting a message data stream from this signal in a noisy channel. From the DOA and the location information within the arriving signal, the WSN has sufficient knowledge to respond to the query of the searching WSN.

B. SIGNIFICANT RESULTS

We investigated the random planar array performance in terms of the half power beam width, the gain of an array of N elements and side lobe performance. The objective was to apply these results to wireless sensor networks to first form a search beam to locate adjacent WSNs, then form a narrow high gain beam to form a communications link. The particular WSNs are assumed to be fixed in location, but the inter-element spacing follows a uniform random distribution within a defined boundary. We demonstrated that we could select a wide beam for the search pattern and simultaneously select the gain. To achieve a given beamwidth, we can choose a physical size, likely a sub-section of the WSN under consideration. Within this physical area, we have control over the gain by choosing the number of elements employed in the array. Alternately, for a given required gain level we can thin the array (or reduce density) without significant loss of gain or the effects of grating lobes. It does not matter which particular sensor nodes within the WSN are used as long as, on average, they are randomly distributed within the physical area chosen. This is important if the objective is to distribute the burden of transmitting, and thus energy consumption among the nodes within the WSN.

We established the need for the receiving WSN to blindly determine the direction of arrival (DOA) of an arriving signal. From the DOA and the range information encoded within the arriving signal, the WSN has sufficient knowledge to build its reply to the query of the searching WSN. Using a random planar array, we utilized a spread spectrum base SDMA technique to determine the DOA of an incident test signal and simulated extracting a message data stream from this signal in a noisy channel with SNRs of 20 db and −5 dB, accurately extracting the data.

C. TOPICS FOR FURTHER RESEARCH

In the process of carrying out this work many issues for further research came to light. This section highlights some of these topics.

In this work, we demonstrated a SDMA method to determine the DOA of a signal in two dimensions assuming vertical polarization. Indeed, the SDMA method shows promise for much greater capabilities. In this thesis, the virtual signals were only used to determine the DOA across the horizon but could be extended to include the hemisphere. In this manner, the graphically depicted UAV of Figure 4 could quickly be located, or in the reciprocal instance, the UAV could quickly find WSNs. Once located, the look directions or virtual signals used for tracking quickly collapses to just a few vectors. This type of three dimensional pattern covering the hemisphere above the array is often referred to in the literature as a "pin cushion" pattern. The SDMA technique can also search for multiple or elliptical polarizations by creating and summing the orthogonal polarization virtual signals [25].

In this work, the spreading was applied only to the receiving WSN to determine the DOA and extract the message. Alternately, the searching WSN could spread its transmitted search signal in the manner described in Chapter IV on SDMA for the receiving array leading to further processing gain.

For the SDMA receiver, we assumed that only the direct signal reached the receiver. For a terrestrial communications link, there would certainly be a multipath signal fading [16]. Future work may investigate the receiver's capability to reject multipath signals and increase the communication link range or bit rate.

In this work, we assumed that the location of each sensor and that the location error could be estimated. Appendix A addresses the methods of self-localization of the array, however the impact of these errors to the SDMA technique were not addressed in this thesis. Future work may determine the impact of localization errors on the SDMA technique. Additionally, future work may address methods for combining the signals received at each of these individual sensor node locations.

The path loss model in this work used a wireless LAN model based on antenna heights of 1 m or above. A better path loss model would give a more accurate estimate of the margin required to form the communications link between two adjacent WSNs. Future work may seek to develop path loss models for ground level transmitter-receiver pairs based on empirical data using actual sensor motes under a variety of operating conditions. Applications for linking together WSNs are only limited by the imagination. The WSNs need not be random or fixed in place on the ground. For example, one WSN may be a periodic array placed on the side of a building and the other could be an array placed on the top or side of a truck. As the truck passes a station, the fixed station would search for the truck's array, establish a communications link and pass information wirelessly to the fixed node. Since the link is established autonomously, the truck doesn't need to stop, or indeed perform any action. A future effort may examine such applications that have potential military value.

APPENDIX A. WIRELESS SENSOR NODE LOCALIZATION

A basic requirement of creating a smart antenna from a WSN is the immediate task of forming a local network of these nodes, identifying each other's location and relative geometry. This map of the geometry is created during the self organization of the WSN. Once the local geometry of the WSN is determined, it will then be possible to form a beam collectively and search for other WSNs. This section deals with determining the local wireless sensor network geometry.

It is assumed that the sensor nodes are deployed or dispersed in a random fashion in the area of interest and that all are properly oriented, i.e. they are not laying on their side or upside down, but have the transmitting antenna pointing vertically. For the purposes of this work, the sensors in a network are assumed to be randomly distributed with a uniform probability density function (pdf).

A. LOCALIZATION METHODS

So important is the need to determine local geometries of an ad hoc wireless network, extensive research has been conducted on this topic and the corresponding literature covers a multiplicity of techniques [4,30,31,32,33]. Here we present a brief description of the methods for determining the geometry of an ad hoc wireless sensor network. Due to the available methodologies and advancing technology, we assume in this thesis that each local WSN is self aware of its absolute location and orientation as well as the geometry of the individual nodes that compose the network.

Self organization allows the WSN to determine its own geometry. These techniques typically estimate distance and angle between nodes, communicate them to the neighboring nodes, then adjust or rotate individual coordinate systems to obtain a single global coordinate system. Magnani and Leung [4] discuss one such algorithm.

Localization methods generally fall into 4 major categories and are described here.

1. GPS Localization

All the sensor nodes have on-board GPS receivers and simply communicate this to adjacent nodes, forming the table of locations and or separation distances.

2. Signal Techniques

These techniques rely on nodes estimating the location of other nodes by measuring a signal parameter [30]. One such method measures the received signal strength as compared to a known transmitted value called Received Signal Strength Indicator (RSSI) or Relative Signal Strength (RSS). Other techniques are Time of Arrival (ToA), Time Difference of Arrival (TDoA) and Angle of Arrival (AoA) estimation. For the AoA case, the sensor node must be equipped with an antenna that is able to measure direction.

3. Anchor Nodes

Anchor nodes or "base stations" of known position are distributed across a sensor network [30,31]. Other nodes' positions are triangulated based on the known locations of the anchors. Careful selection of the base stations' position distribution and precise knowledge of their location are crucial to minimizing position error propagation as subsequent node locations are calculated [31]. Additionally, the triangulation error in locating a particular node can be reduced by dispersing the base stations at the perimeter edges of the sensor field as opposed to picking 3 base stations that are close together.

4. GPS Free Methods

Numerous references are available for the localization of WSN in the absence of GPS data [4,32,33]. These methods require a signal technique and the determination of local and absolute coordinate systems. Reference [31] shows that multi-path errors dominate the measurement noise and subsequently tend to be the main cause of error in range estimation.

Ji and Zha [30] discuss the method of multidimensional scaling as technique to mitigate the problems of cumulative measurement error. This technique builds a series of

local maps of adjacent sensors along a route from a starting anchor to an ending anchor. This method may be of use for the larger problem, discussed in Chapter II on building the OTH network of WSNs.

In this work, we assume the following about the WSNs: the WSN is location aware with respect to its absolute location and orientation; the relative positions of each of the sensor nodes comprising the network are known; the sensor nodes are uniformly distributed within a definable boundary; and the sensor nodes are able to synchronize internal clocks either to a master node or to a GPS reference.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. MATLAB CODE

This appendix contains selected MATLAB code used to generate the figures in this thesis.

A. Comparison of Power Pattern of Linear Array and Average Power Pattern of the Same Size Random Array

This code was used to generate Figure 12.

```
% Chris Taylor Modification to Array factor code presented as a solution to
% Homework 4.1 of below textbook.
% This code produces both the Array factor for a uniform linear array and
% the envelope from a random array of the same size.
% This code also used in the Figure 9 on grating lobes.
% Feb 08    Naval Postgraduate School, Monterey CA
%
% Algorithm for ULA by Frank B. Gross "Smart Antennas for Wireless
% Communications", c 2005, Chapter 4, Homework solutions.
%
% Matlab m-file for Problem 4.1
d = [.5 0.5 0.5]; % d/lambda
%N = [8 8 8]; theta = -pi/2:.01:pi/2;
N = [5 8 20]; theta = -pi:.01:pi;
%----- Create Normalized Array Factors -----%
AFn1 = 1/N(1)*sin(N(1)*pi*d(1)*sin(theta))./sin(pi*d(1)*sin(theta));
AFn2 = 1/N(2)*sin(N(2)*pi*d(2)*sin(theta))./sin(pi*d(2)*sin(theta));
AFn3 = 1/N(3)*sin(N(3)*pi*d(3)*sin(theta))./sin(pi*d(3)*sin(theta));

%----- Linear Array Power Pattern ----------
PAFn1 = AFn1.*conj(AFn1);  %PAFn1=10*log10(abs(PAFn1));
PAFn2 = AFn2.*conj(AFn2);  %PAFn2=10*log10(abs(PAFn2));
PAFn3 = AFn3.*conj(AFn3);  %PAFn3=10*log10(abs(PAFn3));

%---Random Array Predicted Power Pattern Envelope -------%
RAFN1 = (AFn1.*conj(AFn1)).*(1-(1/N(1)))+ 1/N(1)  ;
RAFN2 = (AFn2.*conj(AFn2)).*(1-(1/N(2)))+ 1/N(2)  ;
RAFN3 = (AFn3.*conj(AFn3)).*(1-(1/N(3)))+ 1/N(3)  ;

thetad=theta*180/pi;

%----- Plot Results -----% 4-1
subplot(311), plot(thetad,10*log10(PAFn1),'b',thetad,10*log10(RAFN1),'-c'), grid on
title([ {'N Element Linear Array Power Pattern'},...
        {' vs Average Power Pattern of N Element Random Array'}],...
           'FontSize',14,'FontWeight','bold')
ylabel('\fontsize{14}AF^2(\theta) dB')
xlabel('\fontsize{14}\theta (deg)')
legend('N = 5, d/\lambda = 0.5', 'Random Array Ave PP'),
axis([0 90 -20 0])

subplot(312), plot(thetad,10*log10(PAFn2),'r',thetad,10*log10(RAFN2),'-m'), grid on
ylabel('\fontsize{14}AF^2(\theta) dB')
xlabel('\fontsize{14}\theta (deg)')
```

```
legend('N = 8, d/\lambda = 0.5', 'Random Array Ave PP'),
axis([0 90 -30 0])

subplot(313), plot(thetad,10*log10(PAFn3),'b',...
                   thetad,10*log10(RAFN3),'g'), grid on
ylabel('\fontsize{14}AF^2(\theta) dB')


xlabel('\fontsize{14}\theta (deg)')
legend('N = 20, d/\lambda = 0.5', 'Random Array Ave PP'),
axis([0 90 -40 0])
```

B. Random Planar Array Half Power Beam Width vs Array Size and Random Planar Array: Maximum Gain vs Array Size, Const Number of Elements

This code was used to generate Figures 14 and 16

```
% This code computes far-field gain pattern
% for planar array composed of N vertical half-wave
% dipoles at wavelength normalized coordinates [x(n),y(n)]
% with complex I(n) currents.
%
% Original code for period arrays by M. A. Morgan 29 Oct 07
%
% Modified by C.E. Taylor Dec 07 to use Random Planar Array
% ** This version calculates half power BW vs array size **
%
% This version uses Uniformly current amplitudes (In=1 for all elem)
%
clear all
clf reset

% Setting up 3-D pattern grid
Ntheta=91; Nphi=361;
theta=linspace(0,pi/2,Ntheta); phi=linspace(-pi,pi,Nphi);
[Phi,Theta]=meshgrid(phi,theta);
CT=cos(Theta); ST=sin(Theta); CP=cos(Phi); SP=sin(Phi);
Rx=ST.*CP; Ry=ST.*SP; Rz=CT;  % unit radial vector components

% Half-wave dipole pattern for Iin=1A (eps handles theta=0 indeterminent)
E0 = 60*cos(pi*CT./2)./(ST+eps);

HPBW_ne = zeros(1,50);       % Half Power Beam Width vs  array size
Gain_ne = zeros(1,50);       % Gain vs array size
PeakSLL_ne = zeros(1,50);     % Peak SLL vs size
AveSLL_ne = zeros(1,50);      % Ave SLL vs array size

%------------ Build Random Array ---------------
lambda = 1.0;
k = 2*pi/lambda;
%array_size = 5.0*lambda;   % in wavelengths (lambda) square grid

nruns = 35;
for run=1:nruns;

    fprintf( ' run: %2.0f\n', run);
```

```
for array_size = (1:20)*lambda;
    num_elements = 20;
    Nx = num_elements;
    Ny = num_elements;
    sepn_min = 0;
    xp = array_size * rand(1,num_elements);
    yp = array_size * rand(1,num_elements);

    %------------ Stats of Random Array ---------------

    min_spacing = 0.1*lambda;    % 1/10th wavelength
    while (sepn_min< min_spacing) % min spacing = .1 wavelength

        rp = [xp;yp]';
        sepn = pdist(rp);     % dist between all elements in Random Array
        sepn_min = min(sepn);        % find minimum separation
        zsepn = squareform(sepn);    % convert linear sepn array to square
        [Rmin Cmin] = find(zsepn==sepn_min,1, 'first'); %returns index
%         fprintf(['Min separation is %6.3f wavelengths,'...
%                  'between elements %2.0f and %2.0f \n'],...
%                   sepn_min, Rmin, Cmin)
        if (sepn_min< min_spacing)  % then Move one of them
            xp(Rmin)=array_size * rand;
            yp(Rmin)=array_size * rand;
        end
    end
%     fprintf( 'Element %2.0f Located at %8.3f, %8.3f\n',Rmin,rp(Rmin,:) )
%     fprintf( 'Element %2.0f Located at %8.3f  %8.3f\n',Cmin,rp(Cmin,:) )

    I0=1;  % assuming uniform current amplitudes

    %phid0=input('Enter peak gain azimuth angle in degrees from x-axis: ');
    phid0 = 0;
    phi0=pi*phid0/180;      %convert to radians

    % Phasing I(Nx,Ny) array elements to point array in phi0 direction
    Ip= I0 * exp(-j*k*(cos(phi0)*xp+sin(phi0)*yp));

    % Summing far-field pattern

    F=zeros(size(E0));
    for n=1:num_elements
        xpRx = xp(n)*Rx;
        ypRy = yp(n)*Ry;
        F = F + Ip(n)* exp(j*k*(xpRx+ypRy));
    end

    % Normalized power pattern
    F2 = abs((E0.*F)/(num_elements)).^2;

    DTheta=theta(2)-theta(1); DPhi=phi(2)-phi(1);

    IntF2=DTheta*DPhi*sum(sum((F2).*sin(Theta)));

    Gain=4*pi*F2/IntF2;
    Gmax=max(max(Gain)); GmaxdB=10*log10(Gmax);

    % Horizontal pattern (the x-y plane) and 3dB beamwidth
```

```
    n=find(phi==0);
    m=find(theta==pi/2);
    G3dB=Gmax/2;
    GH=Gain(m,:);
    mf=find(GH >= G3dB,1,'first');
    mp=find(GH >= G3dB,1,'last');
    BWH=180*(phi(mp)-phi(mf))/pi;
    GHdB=10*log10(GH);

    %----------------- calc Side Lobe levels ------------------------
    BW = imregionalmax(GHdB); %MATLAB rtn that finds local maxima->BW array "1"s
    Al = find(BW);            %index pointers to local Maxima
    [bb,ix]=sort(GHdB(find(BW)),'descend');   % sort by highest peak->lowest
    SLL= bb(1)-bb;           % Side Lobe levels, down from peak
    SLi= Al(ix);             % index to peaks
    SLa= 180*phi(SLi)/pi;    % Angle where peaks occur
%     fprintf('First  Sidelobe Level: %6.2f below max at phi= %4.0f \n',SLL(2),SLa(2))
%     fprintf('Second Sidelobe Level: %6.2f below max at phi= %4.0f \n',SLL(3),SLa(3))
%     fprintf('Mean Sidelobe Level: %6.2f below max \n',mean(SLL(2:length(Al))))

    HPBW_ne(array_size) = HPBW_ne(array_size)+BWH ;       % Gain vs number elements array
    theta3db (array_size) = 50.76 / array_size;          % Theory

    Gain_ne(array_size)= Gain_ne(array_size) + GmaxdB;
    PeakSLL_ne(array_size)= PeakSLL_ne(array_size)+SLL(2);     % Peak SLL vs num elements
    AveSLL_ne(array_size)=AveSLL_ne(array_size)+mean(SLL(2:length(Al)));      % Ave SLL
vs num elements

end  % array_size


end  % runs

plot(1:array_size, HPBW_ne(1:array_size)/nruns,'-^', 1:array_size,theta3db)
t1=['Random Planar Array ', num2str(num_elements),' Elements'];
t2='HPBW vs Array Size';
title({t1,t2});
xlabel('Array Size \lambda')
ylabel('Half Power Beam Width ^o')
grid off
legend('Random Array','Analytic')
%axis([ 1 30 0 360]);

figure;
plot(1:array_size, Gain_ne(1:array_size)/nruns,...
     1:array_size, Gain_ne(1:array_size)/nruns - PeakSLL_ne(1:array_size)/nruns,...
     1:array_size, Gain_ne(1:array_size)/nruns - AveSLL_ne(1:array_size)/nruns,...
     1:array_size, Gain_ne(1:array_size)/nruns-13)  % 13 = -10log10(20 elements)
t1=['Random Planar Array ', num2str(num_elements),' Elements'];
t2='Maximum Gain vs Array Size';
title({t1,t2});
axis( [1 20 0 23]);
grid off
xlabel('Array Size \lambda');
ylabel('Power (dB)');
legend('Max Gain', 'Peak SLL', 'Ave SLL','Ave SLL-Theory','Location','NorthWest')
```

C. Random Planar Array: Maximum Gain vs Number of Elements

This code was used in Figure 15.

```
% This code computes far-field gain pattern
% for planar array composed of N vertical half-wave
% dipoles at wavelength normalized coordinates [x(n),y(n)]
% with complex I(n) currents.
%
% By M. A. Morgan 29 Oct 07
%
% Modified by C.E. Taylor Dec 07 to use Random Planar Array
% ** This version calculates gain vs number of elements **
%
% This version uses Uniformly current amplitudes (In=1 for all elem)
%
clear all
clf reset

% Setting up 3-D pattern grid
Ntheta=91; Nphi=361;
theta=linspace(0,pi/2,Ntheta); phi=linspace(-pi,pi,Nphi);
[Phi,Theta]=meshgrid(phi,theta);
CT=cos(Theta); ST=sin(Theta); CP=cos(Phi); SP=sin(Phi);
Rx=ST.*CP; Ry=ST.*SP; Rz=CT;  % unit radial vector components

% Half-wave dipole pattern for Iin=1A (eps handles theta=0 indeterminent)
E0 = 60*cos(pi*CT./2)./(ST+eps);

Gain_ne = zeros(1,50);       % Gain vs number elements array
PeakSLL_ne = zeros(1,50);     % Peak SLL vs num elements
AveSLL_ne = zeros(1,50);      % Ave SLL vs num elements

%------------ Build Random Array ---------------
lambda = 1.0;
k = 2*pi/lambda;
array_size = 5.0*lambda;   % in wavelengths (lambda) square grid
nruns = 75;
for run=1:nruns;

    fprintf( ' run: %2.0f\n', run);

for num_elements = 5:1:50;
    Nx = num_elements;
    Ny = num_elements;
    sepn_min = 0;
    xp = array_size * rand(1,num_elements);
    yp = array_size * rand(1,num_elements);

    %------------ Stats of Random Array ---------------

    min_spacing = 0.1*lambda;    % 1/10th wavelength
    while (sepn_min< min_spacing) % min spacing = .1 wavelength

        rp = [xp;yp]';
        sepn = pdist(rp);     % dist between all elements in Random Array
        sepn_min = min(sepn);        % find minimum separation
        zsepn = squareform(sepn);    % convert linear sepn array to square
        [Rmin Cmin] = find(zsepn==sepn_min,1, 'first'); %returns index
```

```
%         fprintf(['Min separation is %6.3f wavelengths,'...
%                  'between elements %2.0f and %2.0f \n'],...
%                   sepn_min, Rmin, Cmin)
        if (sepn_min< min_spacing)  % then Move one of them
            xp(Rmin)=array_size * rand;
            yp(Rmin)=array_size * rand;
        end
    end
%     fprintf( 'Element %2.0f Located at %8.3f, %8.3f\n',Rmin,rp(Rmin,:) )
%     fprintf( 'Element %2.0f Located at %8.3f  %8.3f\n',Cmin,rp(Cmin,:) )

    I0=1;  % assuming uniform current amplitudes

    %phid0=input('Enter peak gain azimuth angle in degrees from x-axis: ');
    phid0 = 0;
    phi0=pi*phid0/180;      %convert to radians

    % Phasing I(Nx,Ny) array elements to point array in phi0 direction
    Ip= I0 * exp(-j*k*(cos(phi0)*xp+sin(phi0)*yp));

    % Summing far-field pattern

    F=zeros(size(E0));
    for n=1:num_elements
        xpRx = xp(n)*Rx;
        ypRy = yp(n)*Ry;
        F = F + Ip(n)* exp(j*k*(xpRx+ypRy));
    end

    % Normalized power pattern
    F2 = abs((E0.*F)/(num_elements)).^2;

    DTheta=theta(2)-theta(1); DPhi=phi(2)-phi(1);

    IntF2=DTheta*DPhi*sum(sum((F2).*sin(Theta)));

    Gain=4*pi*F2/IntF2;
    Gmax=max(max(Gain)); GmaxdB=10*log10(Gmax);

    % Horizontal pattern (the x-y plane) and 3dB beamwidth
    n=find(phi==0);
    m=find(theta==pi/2);
    G3dB=Gmax/2;
    GH=Gain(m,:);
    mf=find(GH >= G3dB,1,'first');
    mp=find(GH >= G3dB,1,'last');
    BWH=180*(phi(mp)-phi(mf))/pi;
    GHdB=10*log10(GH);

    %----------------- calc Side Lobe levels ------------------------
    BW = imregionalmax(GHdB); %MATLAB rtn that finds local maxima->BW array "1"s
    Al = find(BW);            %index pointers to local Maxima
    [bb,ix]=sort(GHdB(find(BW)),'descend');   % sort by highest peak->lowest
    SLL= bb(1)-bb;           % Side Lobe levels, down from peak
    SLi= Al(ix);             % index to peaks
    SLa= 180*phi(SLi)/pi;    % Angle where peaks occur
%     fprintf('First  Sidelobe Level: %6.2f below max at phi= %4.0f \n',SLL(2),SLa(2))
%     fprintf('Second Sidelobe Level: %6.2f below max at phi= %4.0f \n',SLL(3),SLa(3))
%     fprintf('Mean Sidelobe Level: %6.2f below max \n',mean(SLL(2:length(Al))))
```

```
    Gain_ne(num_elements)= Gain_ne(num_elements) + GmaxdB;
    PeakSLL_ne(num_elements)= PeakSLL_ne(num_elements)+SLL(2);     % Peak SLL vs num
elements
    AveSLL_ne(num_elements)=AveSLL_ne(num_elements)+mean(SLL(2:length(Al)));      % Ave
SLL vs num elements


end  % num_elements


end  % runs

ne = 1:num_elements;
Gain_ne=Gain_ne/nruns;
PSSL = Gain_ne-PeakSLL_ne/nruns;     % Peak SLL actual
ASSL = Gain_ne-AveSLL_ne/nruns;      % Ave SLL actual
Gth= Gain_ne+ 10*log10(1./ne);       % Theory ASLL (Steinberg)
Gain_th2 = 10*log10(ne)+10*log10(1.6); % Theory: Gain of N dipoles, incl gain of dipole

plot(ne, Gain_ne, ne,PSSL,...
    ne,ASSL, ne,Gth,  ne, Gain_th2  )
t1=['Random Planar Array ', num2str(array_size),' \lambda square'];
t2='Maximum Gain vs Number of Elements';
title({t1,t2});
axis( [5 50 0 23]);
xlabel('Number of Elements in Array');
ylabel('Power (dB)');
legend('Gain', 'Peak SLL', 'Ave SLL', 'Theory-Ave SLL',...
     'Theory-N dipoles', 'Location','NorthWest')
grid off
```

D. DOA determination for the Uniform Linear Array, with two incident signals.

This code used in Figure 23, as an example of the SDMA technique.

```
%  Modified by Chris Taylor Jan 2008 to include pn seq generation or Walsh
%  codes, samples per chip, and two incident beams

%  This is a simulation to model the Elam SDMA technique
%  Developed by Frank B. Gross, November 11, 2004

% define the sample baseband modulation initially as a 1kHz phase modulation tone
% WALSH FUNCTIONS WORK BETTER THAN PN SEQUENCES,Can also use Gold codes
clear; clc;

f=1E3;
T=1/(4*f);
Nchips=64;
SPC = 4;           % Samples Per Chip  (4 min)
tend=SPC*Nchips;    % time snapshot end
t=[0:(tend-1)]*T/(tend-1);
zeta=sin(2*pi*f*t);   % modulation m
%m=zeros(1,length(m));

% plot array factor for linear array
N=11;  % number of receive elements to give 10 degree beamwidth
```

```
d=.5;  % spacing in wavelengths between adjacent array elements

% generate pseudo-random chip codes T long to modulate the
% received signals from each antenna

%Construct a PN object usnign MATLAB pn seq gen
% h = seqgen.pn('Shift', 0);
% % Output chipped PN bits
% set(h, 'NumBitsOut', Nchips);
% for i=1:N
%    B(i,:) = generate(h);
% end
% B = Dmap(B,1);  % map 0,1 to +-1

% use Walsh Sequence instead of pure pn
H = hadamard(Nchips);
for i=1:N
     B(i,:)=H(i+1,:);   % don't want 1st row
end

% spread B by SPC
B = repmat( B,SPC,1);         % repeat by spread factor
B = reshape(B,N,SPC*Nchips);  % like chips together
B = B*pi/2;                   % pi/2 since inside exponential

%test SDMA concept by elimnating the pn sequence
%B=zeros(N,SPC*Nchips);

test=-30*pi/180;
test2=45*pi/180;
Vr=zeros(361,length(t));
Vr2=zeros(361,length(t));
er=zeros(361,length(t));
for k=1:361;
   an(k)=(k-1)-180;
   th(k)=an(k)*pi/180;

  % calculate the actual received signal for each angle using the N element
  % receiver and sum all N signals.
   for i=1:N                % create combined signal at angle k
      Vr(k,:)= Vr(k,:)+ exp(1j*(zeta+2*pi*(i-1)*d*sin(test) +B(i,:)));
      er(k,:)= er(k,:)+ exp(1j*(     2*pi*(i-1)*d*sin(th(k))+B(i,:)));
      Vr2(k,:)=Vr2(k,:)+exp(1j*(zeta+2*pi*(i-1)*d*sin(test2)+B(i,:)));

   end
   Vr(k,:)=Vr(k,:)+Vr2(k,:);

   R1(k)=sum(Vr(k,:).*     er(k,:));
   R2(k)=sum(Vr(k,:).*conj(er(k,:)));
   phase1(k,:)=unwrap(angle(R1(k)));
   phase2(k,:)=unwrap(angle(R2(k)));
end
AF=sin(N*pi*d*(sin(th)-sin(test))+eps)./(N*sin(pi*d*(sin(th)-sin(test)))+eps);
AF2=sin(N*pi*d*(sin(th)-sin(test2))+eps)./(N*sin(pi*d*(sin(th)-sin(test2)))+eps);
maxAF=max(abs(AF));
maxAF2=max(abs(AF2));
maxR2=max(abs(R2));
maxR1=max(abs(R1));
figure;
```

```
plot(an,abs(R2)/maxR2,'b',an,abs(AF+AF2),'g')
% plot(an,abs(R2)/maxR2,'b',an,abs(AF),'g',an,abs(AF2),'r')
xlabel('\theta')
ylabel('Correlation Magnitude')
titlestring0 = ['Uniform Linear Array - SDMA'];
titlestring=['N=',num2str(N),' d=\lambda/2,','\theta inc = '...
    ,num2str(test*180/pi),',',num2str(test2*180/pi) ];
titlestring2=['Nchips=',num2str(Nchips),' Samples/chip =', num2str(SPC)];
title({titlestring0,titlestring,titlestring2})
axis([-90 90 0 1])
legend('Rk','Array Factor');
grid on

% figure;
% titlestring=['Correlation Comparison \theta_k =',num2str(test*180/pi)];
% titlestring2='comparison of instantaneous and averaged phases';
% plot(t,zeta,t,phase2(121,:))
% xlabel(‘time (t)')
% ylabel('phase')
% axis( [0 t(length(t)) -2 2])
% grid on
% title({titlestring,titlestring2});
% legend('phase zeta(t)','average phase')
```

E. SDMA Technique Applied to the Random Planar Array

This code was used to generate Figure 24 the Signal DOA determination for a random planar array.

```
% Chris Taylor Modification to Dr. Gross' SDMA sim
% to allow for Random Planar array vice ULA
% Feb 08    Naval Postgraduate School, Monterey CA
%
% Algorithm from Frank B. Gross "Smart Antennas for Wireless
% Communications", c 2005, Chapter 8, p253-260
%
%  This is a simulation to model the Elam SDMA technique
%  Developed by Frank B. Gross, November 11, 2004
%  WALSH FUNCTIONS WORK BETTER THAN PN SEQUENCES
%  Can also use Gold codes
clear; clc; clf reset;

% ------------- Define Constants -----------------

% define the sample baseband modulation initially as a 300MHz phase modulation tone
f=3E8;                          % freq of baseband modulation
T=1/(4*f);                      % time bandwidth product = 4
Nchips=64;                      % Number of Chips / Time seq
SPC = 16;                       % Samples Per Chip
num_elements = 30;              % number of antenna elements
phi_inc=30*pi/180;              % incident signal of interest

lambda = 3E8/f;                 % c/f
k = 2*pi/lambda;                % wavenumber

tend=SPC*Nchips;                % time snapshot end
t=[0:(tend-1)]*T/(tend-1);      % discritization of signal
zeta=sin(2*pi*f*t);                % modulation zeta (the signal phase sent)
```

```
Ntheta=91; Nphi=361;           % Set up angle matrices
%Ntheta=5; Nphi=13;
theta = linspace(0,pi/2,Ntheta);
phi   = linspace(-pi,pi,Nphi);
DTheta= theta(2)-theta(1); DPhi = phi(2)-phi(1);
[Phi,Theta] = meshgrid(phi,theta);
CT=cos(Theta); ST=sin(Theta); CP=cos(Phi); SP=sin(Phi);
Rx=ST.*CP; Ry=ST.*SP; Rz=CT;  % unit radial vector components

%  ------------ generate pseudo-random chip codes -------------

% T long to modulate the received signals from each antenna
% "codes" here = "Beta" nomenclature from text
H = hadamard(Nchips);   % use Walsh Sequence
for i=1:num_elements
    codes(i,:)=H(i+1,:);   % don't want 1st row
end
% spread codes by SPC
codes = repmat( codes,SPC,1);         % repeat by spread factor
codes = reshape(codes,num_elements,SPC*Nchips);  % move like chips together
codes = codes*pi/2;                   % pi/2 since inside exponential

%_____________________________________________________________________
%
%             ------------ Build Random Array ---------------
%_____________________________________________________________________

array_size = 5.0*lambda;   % in wavelengths (lambda) square grid
min_spacing = 0.1*lambda;  % 1/10th wavelength
sepn_min = 0;              % initialize prior to 1st test
xp = array_size * rand(1,num_elements);
yp = array_size * rand(1,num_elements);

%------------ Stats of Random Array ---------------

while (sepn_min< min_spacing) % min spacing = .1 wavelength

    rp = [xp;yp]';
    sepn = pdist(rp);     % dist between all elements in Random Array
    sepn_min = min(sepn);        % find minimum separation
    zsepn = squareform(sepn);    % convert linear sepn array to square
    [Rmin Cmin] = find(zsepn==sepn_min,1, 'first'); %returns index
    fprintf(['Min separation is %6.3f wavelengths,'...
             'between elements %2.0f and %2.0f \n'],...
              sepn_min, Rmin, Cmin)
    if (sepn_min< min_spacing)  % then Move one of them
        xp(Rmin)=array_size * rand;
        yp(Rmin)=array_size * rand;
    end
end
fprintf( 'Element %2.0f Located at %8.3f, %8.3f\n',Rmin,rp(Rmin,:) )
fprintf( 'Element %2.0f Located at %8.3f  %8.3f\n',Cmin,rp(Cmin,:) )

%------------ Plot Random Array ------------------
subplot(2,2,1)
set(gcf, 'PaperOrientation', 'portrait');
plot(xp,yp,'MarkerSize',15,'Marker','*','LineWidth',2,...
           'LineStyle','none','Color',[0 1 0]);
```

```
tstrg = 'Sensor Cluster, Uniform Excitation';
tstrg2= [num2str(num_elements),' Elements'];
title({tstrg,tstrg2}, 'FontSize',14)
axis([0 array_size 0 array_size])
axis square
set(gca,'XTick',linspace(0,array_size,6))
set(gca,'YTick',linspace(0,array_size,6))
xlabel('x - wavelengths')
ylabel('y - wavelengths')
grid on;
for i = 1:num_elements;
    text(xp(i),yp(i), num2str(i), 'FontSize',10)
end

%------------- plot coord system ---------------
subplot(2,2,2)
coord_syst_diag(1,phi_inc*180/pi,90,array_size)
plot3(xp-array_size/2,yp-array_size/2,zeros(length(xp)),...
    'Marker','*','LineStyle','none','Color',[0 1 0]);
plot3(CP(1,:),SP(1,:), zeros(size(CP(1,:))),'c')
title('Coordinate System and Scan Plane');

%_____________________________________________________________________
%
%        Calc Array Factor for Planar Random Array
%_____________________________________________________________________


% Half-wave dipole pattern for Iin=1A (eps handles theta=0 indeterminent)
E0 = 60*cos(pi*CT./2)./(ST+eps);
I0=1;                           % assuming uniform current amplitudes
Ip= I0 * exp(-j*k*(cos(phi_inc)*xp+sin(phi_inc)*yp));

% Summing far-field pattern

AF=zeros(size(E0));
for n=1:num_elements
    xpRx = xp(n)*Rx;
    ypRy = yp(n)*Ry;
    AF = AF + Ip(n)* exp(j*k*(xpRx+ypRy));
end

%_____________________________________________________________________
%
%                   Calc Received Signals via SDMA
%_____________________________________________________________________

yr=zeros(Nphi,length(t));
ye=zeros(Nphi,length(t));
m=find(theta==pi/2);          % want row for xy plane
Rxm = Rx(m,:);                % 1xNphi vector
Rym = Ry(m,:);

for p=1:Nphi;

  % calculate the actual received signal for each angle using the N element
   % receiver and sum all N signals.
   for n=1:num_elements
      xpRx = xp(n)*Rxm(p);
```

```
      ypRy = yp(n)*Rym(p);
      re = cos(phi_inc)*xp(n)+sin(phi_inc)*yp(n);
      yr(p,:)=yr(p,:)+exp(j*(zeta+ k*(xpRx+ypRy) +codes(n,:)));
      ye(p,:)=ye(p,:)+exp(j*(      k*(re)        +codes(n,:)));
   end

   R(p)=sum(yr(p,:).*conj(ye(p,:)));
end
maxR=max(abs(R));

%_____________________________________________________________________________
%
%                   Calc Gain, Side Lobe Levels, HPBW, etc.
%_____________________________________________________________________________

% Horizontal pattern (the x-y plane) and 3dB beamwidth
n=find(phi==0);
m=find(theta==pi/2);

F2 = abs((E0.*AF)/(num_elements)).^2;  % Normalized power pattern
IntF2=DTheta*DPhi*sum(sum((F2).*sin(Theta)));
Gain=4*pi*F2/IntF2;
Gmax=max(max(Gain)); GmaxdB=10*log10(Gmax);

% Convert Correlation Magnitude to Gain Pattern
RF2 = abs((E0(m,:).*R)/(num_elements)).^2;
GainR2 = 4*pi*RF2/IntF2;   %use same normalizing factor
GRdB = 10*log10(GainR2);

% Steinberg's Random element Gain pattern Eqn 8.10
F2_Rand_ave = F2.*(1-1/num_elements)+1/num_elements;
Gain_Rand_ave = 4*pi*F2_Rand_ave/IntF2;
GRA_dB=10*log10(Gain_Rand_ave(m,:));


G3dB=Gmax/2;
GH=Gain(m,:);
mf=find(GH >= G3dB,1,'first');
mp=find(GH >= G3dB,1,'last');
BWH=180*(phi(mp)-phi(mf))/pi;
GHdB=10*log10(GH);
AFmax = max(abs(AF(m,:)));

%----------------- calc Side Lobe levels ------------------------
BW = imregionalmax(GHdB); %MATLAB rtn that finds local maxima->BW array "1"s
Al = find(BW);             %index pointers to local Maxima
[bb,ix]=sort(GHdB(find(BW)),'descend');   % sort by highest peak->lowest
SLL= bb(1)-bb;           % Side Lobe levels, down from peak
SLi= Al(ix);             % index to peaks
SLa= 180*phi(SLi)/pi;    % Angle where peaks occur
fprintf('First  Sidelobe Level: %6.2f below max at phi= %4.0f \n',SLL(2),SLa(2))
fprintf('Second Sidelobe Level: %6.2f below max at phi= %4.0f \n',SLL(3),SLa(3))
fprintf('Mean Sidelobe Level: %6.2f below max \n',mean(SLL(2:length(Al))))


%_____________________________________________________________________________
%
%                   Plot Array Factor vs. SDMA Pattern
%_____________________________________________________________________________
```

```
% subplot(2,2,3:4)
% plot(phi*180/pi,abs(R)/maxR,'b',phi*180/pi,abs(AF(m,:))/AFmax,'g')
% xlabel('\phi')
% ylabel('Correlation Magnitude')
% tstring=['Number Elements=',num2str(num_elements),...
%          ' Nchips=',num2str(Nchips)];
% tstring2=['\phi inc = ',num2str(phi_inc*180/pi), '  M=', num2str(length(s))];
% title({tstring,tstring2},'FontSize',14)
% axis([-180 180 0 1])
% legend('Rk','Array Factor');
% grid on


%_____________________________________________________________________________
%
%                   Plot Gain vs. SDMA Gain Pattern
%_____________________________________________________________________________

GHMax=max(GHdB);
GRMax=max(GRdB);
GRAMax=max(GRA_dB);
Diff=GRMax-GHMax;   % SDMA only calculated in one planar cut vice sphere,...
                    % Normalization factor is adjusted to acct for diff.
phid=phi*180/pi;

subplot(2,2,3:4)
plot(phid,GRdB-Diff,'b',phid,GHdB,'g','LineWidth',1.5)
%plot(phid,GHdB,'b',phid,GRdB-Diff,'g',phid,GRA_dB,'r','LineWidth',2)
v=axis; v(1)=-180; v(2)=180; if v(3) < -60, v(3)=-60; end
axis(v)
t1 = ['Horizontal Plane: G_{max} = ',num2str(GmaxdB,'%8.1f'),'dB'];
t2 = ['HPBW= ', num2str(BWH,'%8.1f'),' deg',...
      '  SLL=', num2str(SLL(2),'%6.2f below Peak')];
title({t1,t2},'FontSize',14)
xlabel('\phi (deg)','FontSize',14)
ylabel('Gain Pattern (dB)','FontSize',14)
legend('SDMA Correlation', 'Planar Array Gain');
grid on
```

F. Incident Signal Phase, Walsh-Hadamard Codes & Received Signal

This code used to generate Figure 22 showing the received signal phase, Walsh-Hadamard Spreading Codes and the real part of the chipped signal r(t).

```
%  This is a simulation to model the Elam SDMA technique
%  Developed by Frank B. Gross, November 11, 2004

% define the sample baseband modulation initially as a 1kHz phase modulation tone
% WALSH FUNCTIONS WORK BETTER THAN PN SEQUENCES,Can also use Gold codes

%  Modified by Chris Taylor Jan 2008 to include pn seq generation or Walsh
%  codes, samples per chip, and two incident beams

%  This short code shows the phase of incident signal zeta(t), Walsh codes
%  and the Real part of the received signal r_n(t)

clear; clc;

f=1E3;
T=1/(4*f);
Nchips=64;
SPC = 4;           % Samples Per Chip  (4 min)
tend=SPC*Nchips;    % time snapshot end
t=[0:(tend-1)]*T/(tend-1);
zeta=sin(2*pi*f*t);   % modulation m

% plot array factor for linear array
N=11;  % number of receive elements to give 10 degree beamwidth
d=.5;  % spacing in wavelengths between adjacent array elements

% generate pseudo-random chip codes T long to modulate the
% received signals from each antenna

% use Walsh Sequence instead of pure pn
H = hadamard(Nchips);
for i=1:N
     B(i,:)=H(i+1,:);   % don't want 1st row
end

% spread B by SPC
B = repmat( B,SPC,1);         % repeat by spread factor
B = reshape(B,N,SPC*Nchips);  % like chips together
B = B*pi/2;                   % pi/2 since inside exponential

figure;
subplot(3,1,1)
plot(t,zeta,'b')
axis( [0 T -2 2])
xlabel('time (t)')
ylabel('Phase - \zeta(t)')
grid on

subplot(3,1,2)
plot(t,B(11,:),'r')           % use arbitrary WH row
axis( [0 T -2 2])
xlabel('time (t)')
ylabel('Walsh-Hadamard Code')
```

```
grid on

subplot(3,1,3)
plot(t,real(exp(j*(zeta+B(11,:)))),'m')
axis( [0 T -2 2])
xlabel('time (t)')
ylabel('Real Part of r_n(t)')
grid on
```

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

[1] Crossbow Technologies home page, March 2008, http://www.xbow.com/Home/HomePage.aspx.

[2] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, "A Survey of sensor networks," *IEEE Communications Magazine*, Volume 40, Issue 8, pp 102-114, August 2002.

[3] smartBridges Inc, Customer Success Stories website, "Wireless Farm Area Network for Rice Field Research," April 2008, http://www.smartbridges.com/css/articles.asp?id=491.

[4] A. Magnani and K.K. Leung, "Self-Organized, Scalable GPS-Free Localization of Wireless Sensors", in *Wireless Communications and Networking Conference, 2007.WCNC 2007. IEEE*, Kowloon, Hong Kong, 11-15 March 2007 pp 3798-3803.

[5] P. Vincent**,** "Energy conservation in wireless sensor networks", PhD dissertation, Naval Postgraduate School, Monterey, California, USA, June 2007.

[6] C. W. Chan, "Distributed beamforming in wireless sensor networks", Masters thesis, Naval Postgraduate School, Monterey, California, USA, December 2004.

[7] J. Litva, T. K. Lo, *Digital Beamforming in Wireless Communications,* Artech House, Boston, 1996.

[8] C. H. M. Tong, "System study and design of broad-band U-slot microstrip patch antennas for aperstructures and opportunistic arrays," M.S. thesis, Naval Postgraduate School, Monterey, California, USA, December 2005.

[9] M. Batson, Enhanced radio frequency (rf) collection with distributed wireless sensor networks," PhD dissertation, Naval Postgraduate School, Monterey, California, USA, June 2007.

[10] L. C. Godara, "Application of antenna arrays to mobile communications, part II: beam-forming and direction-of-arrival considerations," in *Proceedings of the IEEE*, Vol. 85, No. 8, pp 1195-1245, August 1997.

[11] Y.-W. Hong, W.-J. Huang, F.-H. Chiu, and C.-C. Jay Kuo, "Cooperative Communications in Resource-Constrained Wireless Networks," *IEEE Signal Processing Magazine*, Vol. 24, Issue 3, pp 47-57, May 2007.

[12] B. Sklar, *Digital Communications Fundamentals and Applications*, Prentice Hall, New Jersey, 2001, p 255.

[13] Crossbow®web site link, February 2008, http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICAz_Datasheet.pdf .

[14] D. B. Green and M.S Obaidat, “An accurate line of sight propagation performance model for ad-hoc 802.11 wireless lan (WLAN) devices,” in *IEEE International Conference on Communications 2002*, New York, NY, Vol 5, pp 3424-3428, 28 April-2 May 2002.

[15] Y. Okumura et al, “Field strength and its variability in UHF and VHF land-mobile radio service,” *Review of the Electical Communications Laboratories*, Vol. 16, 1968, pp 825-873.

[16] M. Hata, “Emperical formula for propagation loss in land mobile radio services,” *IEEE Transactions on Vehicular Technology,* Vol. VT-29, No. 3, pp 317-325, August 1980.

[17] JPL’s Wireless Communications Reference Website, March 2008, http://wireless.per.nl/reference/contents.htm.

[18] W. Stutzman & G. Thiele, *Antenna Theory and Design*, John Wiley & Sons, New York, 1981.

[19] B. D. Steinberg, *Principles of Aperture and Array System Design, including Random and Adaptive Arrays*, John Wiley & Sons, New York, 1976, p 77.

[20] F. Gross, *Smart Antennas*, McGraw Hill, New York, 2005, p 92.

[21] V. Rodoplu and T. Meng, ”Minimum Energy Mobile Wireless Networks,” in *IEEE Journal on Selected Areas in Communications*, Vol. 17, No. 8, pp. 1333-1344, August 1999.

[22] R.W. Sinnott, "Virtues of the Haversine", *Sky and Telescope*, vol. 68, no. 2, 1984, p. 159.

[23] Movable Type Ltd, http://www.movable-type.co.uk/scripts/latlong.html, March 17, 2008

[24] M. L. Skolnik, *Introduction to Radar Systems, 3rd ed.*, Tata McGraw-Hill, New Delhi, 2001, pp 496-7.

[25] Elam, C. “Method and Apparatus for Space Division Multiple Access Receiver,” Patent No. 6,823,021, Rights assigned to Greenwich Technology Associates, One Soundview Way, Darien, CT. November 23, 2004.

[26] R. L. Peterson, R. E. Ziemer, D. E. Borth, *Introduction to Spread Spectrum Communications*, Prentice Hall, Upper Saddle River, NJ, 1995, pp 113-135.

[27] J. G. Proakis, *Digital Communications, Fourth Edition*, McGraw Hill, New York, 2001, pp 744-745.

[28] H. D. Luke, “Sets of one and higher dimensional Welti codes and complementary codes,” in *IEEE Transactions on Aerospace and Electronic Systems, Vol AES-21, No. 2. March 1985*, pp 170-179.

[29] Dr. F. Gross, (personal communication) February 5, 2008.

[30] Xiang Ji and Hongyuan Zha, "Sensor positioning in wireless ad-hoc sensor networks using multidimensional scaling," in *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, Hong Kong, PR China, volume 4, March 7-11, 2004, pp 2652 – 2661.

[31] Z. Chaczko, et al. , "Methods of sensors localization in wireless sensor networks," in *Proceedings of the 14th Annual IEEE International Conference and Workshops on the Engineering of Computer Based Systems*, Tuscon, AZ, August 2007, pp 145-152.

[32] S. Capkun, et al., "GPS-free positioning in mobile Ad-Hoc networks," Proceeding sof the 34th Hawaii International Conference on System *Sciences*, Maui, HI, September 2001, pp 1-10.

[33] R. Iyengar and B. Sikdar, "Scalable and Distributed GPS free Positioning for Sensor Networks," *IEEE International Conference on Communications, 2003, ICC '03*, Anchorage, AK, Volume 1, 11-15 May 2003, pp 338-342.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Prof. Murali Tummala
   Naval Postgraduate School
   Monterey, California

4. Prof. John McEachen
   Naval Postgraduate School
   Monterey, California

5. Chris Taylor
   Naval Postgraduate School
   Monterey, California